LOAN DOCUMENT

PHOTOGRAPH THIS SHEET

LEVEL

INVENTORY

0

DTIC ACCESSION NUMBER

WL-TR-95-4067

DOCUMENT IDENTIFICATION

May 95

DISTRIBUTION STATEMENT

ACCESSION FOR

NTIS GRA&I
DTIC TRAC
UNANNOUNCED
JUSTIFICATION

BY

DISTRIBUTION/

AVAILABILITY CODES

DISTRIBUTION	AVAILABILITY AND/OR SPECIAL	
A-1		

DISTRIBUTION STAMP

DTIC
S ELECTE D
AUG 2 1995
C

DATE ACCESSIONED

H A N D L E W I T H C A R E

DATE RETURNED

19950731 106

DATE RECEIVED IN DTIC

REGISTERED OR CERTIFIED NUMBER

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-FDAC

DTIC FORM JUN 90 70A

DOCUMENT PROCESSING SHEET

PREVIOUS EDITIONS MAY BE USED UNTIL STOCK IS EXHAUSTED.

LOAN DOCUMENT

WL-TR-95-4067

ELECTRONIC PROTOTYPING REVIEW MEETING

TECHNICAL MANAGEMENT CONCEPTS INC
PO BOX 340345
BEAVERCREED, OH 45434-0345

MAY 1995

FINAL REPORT FOR 05/01/95-05/31/95

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.

MATERIALS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT PATTERSON AFB OH 45433-7734

REPORT DOCUMENTATION PAGE

FORM APPROVED
OMB NO. 0704-0188

1. AGENCY USE ONLY *(Leave Blank)*

2. REPORT DATE
May 1995

3. REPORT TYPE AND DATES COVERED
May 95 - May 95

4. TITLE AND SUBTITLE
Electronic Prototyping Review Meeting

5. FUNDING NUMBERS
C: F33615-94-D-5801
PE: 62102F
PR: 2418
TA: 90
WU: 01

6. AUTHOR(S)
See Agenda

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Technical Management Concepts Inc
PO Box 340345
Beavercreek, OH 45434-0345

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Materials Directorate
Wright Laboratory
Air Force Materiel Command
Wright Patterson AFB OH 45433-7734

10. SPONSORING/MONITORING AGENCY REP NUMBER
WL-TR-95-4067

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release: distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT
This document is a compendium of the presentations that were made during the Electronic Prototyping Review Meeting that was held on May 23 and 24, 1995 in Dayton, Ohio. The review was sponsored by the Materials Process Design Branch of the Air Force Wright Laboratory Materials Directorate and the Air Force Office of Scientific Research. The meeting addressed progress that has been made in basic research in the area of computational methods for modeling and analysis of geometry and applied research in the area of Materials Process Design, specifically the mapping of form and function to materials and processes. Topics addressed are:

- Soft Optimization of Hard Problems.
- Process Model Development for use with Discrete Event System Techniques.
- Analyzing Discrete Event Simulation Models of Complex Manufacturing Systems: A Computational Complexity Approach.
- Finite Element Analysis for Simultaneous Design for Deformation Processes.
- Automated 3-D Mesh Generation for Deformation Process Design.
- Alternative Materials & Processes: Incomplete Design and Multi-Process Optimization.
- An Evolving Computational Model for In-situ Material Aging Assessment.
- Materials Design using Pyramidal Networks.
- Instant Learning for Functional Approximation.
- An Engineering Approach to Surface Blending using Pythagorean Hodographs.
- A Constraint-Driven 'Piping-Algorithm' using Cyclides.

DTIC QUALITY INSPECTED 5

14. SUBJECT TERMS
Computational Modeling, Process Design, Process Modeling, Finite Element Analysis, Incomplete Design, Simultaneous Design, Process Optimization, Soft Optimization, Materials Design, Pyramidal Networks, Surface Blending, Pythagorean Hodographs, Piping Algorithm, Cyclides

15. NUMBER OF PAGES
361

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASS OF THIS PAGE.	19. SECURITY CLASS OF ABSTRACT	20. LIMITATION ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std Z39-18
298-102

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Steven R. LeClair

STEVEN R. LECLAIR, Chief
Materials Process Design
Integration & Operations Division
Materials Directorate

Steven R. LeClair

STEVEN R. LECLAIR, Chief
Materials Process Design
Integration & Operations Division
Materials Directorate

John R. Williamson

JOHN R. WILLIAMSON
Integration & Operations Division
Materials Directorate

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WL/MLIM, Wright Patterson AFB, OH 45433 to help maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

Optimization and Discrete Mathematics
Air Force Office of Scientific Research
(from AFOSR Pamphlet 70-1, 1 October 1994)

Our goal is to develop mathematical methods for solving large or complex problems, such as those occurring in logistics, engineering, design, or strategic planning. These problems can often be formulated as mathematical programs. Therefore, research is directed at linear and nonlinear programming methods, especially those that can be implemented on parallel computers. We are also emphasizing discrete structures, as they often represent important Air Force problems.

Three areas of particular importance are emphasized in discrete mathematics. First is the optimal solution of integer programming models and other combinatorially based structures. These structures arise in areas of interest to the Air Force, such as design of very-large-scale integrated networks, frequency assignment, and scheduling and routing. Second, in addition to the evolution of traditional solution methods, the program supports new algorithmic paradigms such as simulated annealing and genetic algorithms. Third, we support research in computational geometry, especially as it relates to electronic prototyping.

Research in optimization focuses on the development of special algorithms for the particular structures that arise, emphasizing implementation on parallel architectures. Since networks are so important for military logistics problems, optimization over networks is a major component of our program. Some research on stochastic optimization, which will benefit from increased parallelism, will begin, as will research on the use of nonlinear programming for the optimization of polymers and biomolecules.

Dr Neal Glassman, AFOSR/NM
(202) 767-5027; DSN 297-5027
FAX: (202) 404-7496

Electronic Prototyping Review Meeting

May 23-24, 1995
Wright State University, Dayton, Ohio

May 22:

6:00 - 8:00 PM Social at Holiday Inn in 1st Floor Hospitality Suite

May 23:

8:30 Welcome
Dean Brandeberry, Wright State University

8:45 Opening Remarks: Electronic Prototyping
S. LeClair, Wright Laboratory

9:00 Soft Optimization of Hard Problems (Hard Copy) ✓
Y.C. Ho, Harvard University

9:45 - 10:00 Break

10:00 Process Model Development for Use with Discrete Event System Techniques
J. S. Gunasekera, Wright Laboratory ✓

10:45 Analyzing Discrete Event Simulation Models of Complex Manufacturing Systems: A Computational Complexity Approach
S. H. Jacobson & Capt A.W. Johnson, Virginia Tech ✓

12:00-1:30 Lunch

1:30 Finite Element Analysis for Simultaneous (Billet, Die and Part) Design for Deformation Processes
T. Hughes. Stanford ✓

2:15 Automated 3-D Mesh Generation for Deformation Process Design
M. Shepherd, RPI ✓

3:00 Alternative Materials & Processes: Incomplete Design and Multi-process Optimization
A. Chemaly, Wright Laboratory

3:45-4:00 Break

4:00 Incomplete Design for Machining Demonstration
H. AL-Kamhawi, Wright Laboratory

May 24:

8:30 An Evolving Computational Model for In-situ Material Aging Assessment
Y. Pao, Wright Laboratory

9:15 Materials Design using Pyramidal Networks
A. Jackson, Wright Laboratory

10:00 - 10:15 Break

10:15 A Rapid Supervised Learning Neural Network for Function Interpolation and Approximation
P. Chen, Wright Laboratory

11:00 Constant-radius Blending of Free-Form Parametric Surfaces and New Results on Pythagorean-Hodograph Curves
R. Farouki, University of Michigan

12:00 - 1:30 Lunch

1:30 A Constraint-Driven 'Piping-Algorithm' using Cyclides
D. Dutta, University of Michigan

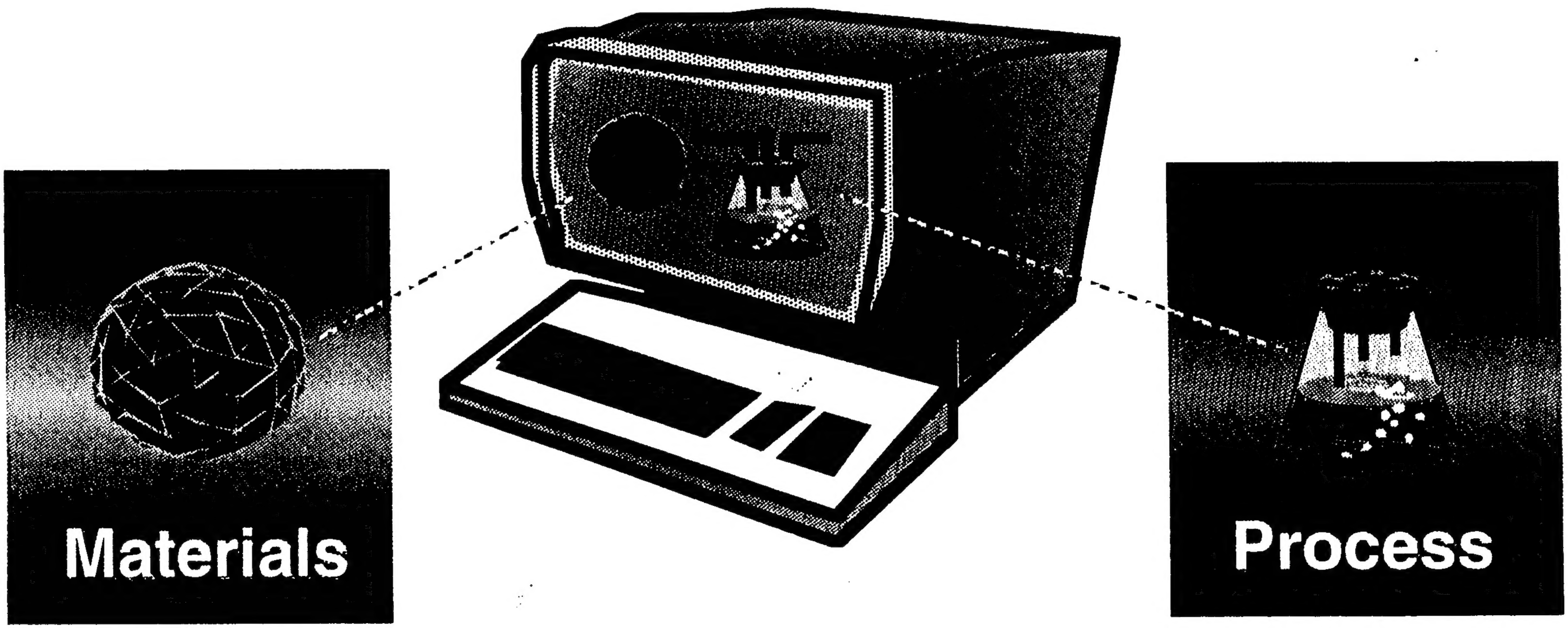
2:30-4:00 Forum Discussion:

User Perspectives on Electronic Prototyping Needs.

Youngstown State University , Technology Development Consortia (TDC)
San-Antonio Air Logistic Center (SA-ALC)
Sacramento Air Logistic Center (SM-ALC)
Oklahoma City Air Logistic Center (SA-ALC)
Systems Research Laboratories (SRL)
Allison Engines
BF Goodrich (Wheel & Brake Division)
Lockheed-Martin
Cherry Point Naval Depot
General Motors Corporation - Saginaw Division
Tennessee Technological University - Center for Mfg Research
Southern Illinois University

Electronic Prototyping

Computational Methods for Design



Air Force Office of Scientific Research
&
Wright Laboratory, Materials Directorate

4

Electronic Prototyping

Computational Methods for Design

Initiative Duration: 1991 - Present

AFOSR Mgr: Dr Neal Glassman (Math Directorate)

Objective: Basic Research in the area of computational methods for modeling and analysis of geometry.

Materials Directorate Principal Investigator: Dr Steve LeClair

Objective: Applied Research (leverage Basic Research) in the area of Materials Process Design, specifically the mapping of form & function to materials & processes.

Universities who have been AFOSR Grant Funded:

Cornell, Purdue, UMich, Harvard, & Virginia Tech

University Visiting Scientist Participants who have been WL Funded:

Case Western Reserve, Wright State, Ohio State, UCinn.

Materials & Process Challenge

"It is estimated that the amount of information in the world *(currently)* doubles every 20 months"

[redacted]

<u>Example:</u> "It has been estimated that in 1965, a mechanic who understood about 500 pages of various repair manuals could fix just about any car on the road. Today, that same mechanic would need nearly 500,000 pages of manuals - equivalent to roughly 50 New York City telephone books"

[redacted]

AIR FORCE NEED:

In addition to using past experience to avoid re-learning how to: design, manufacture and repair our defense systems,

we must also find ways to automate the organization of <u>new</u> experience(s).

6

Materials Process Design

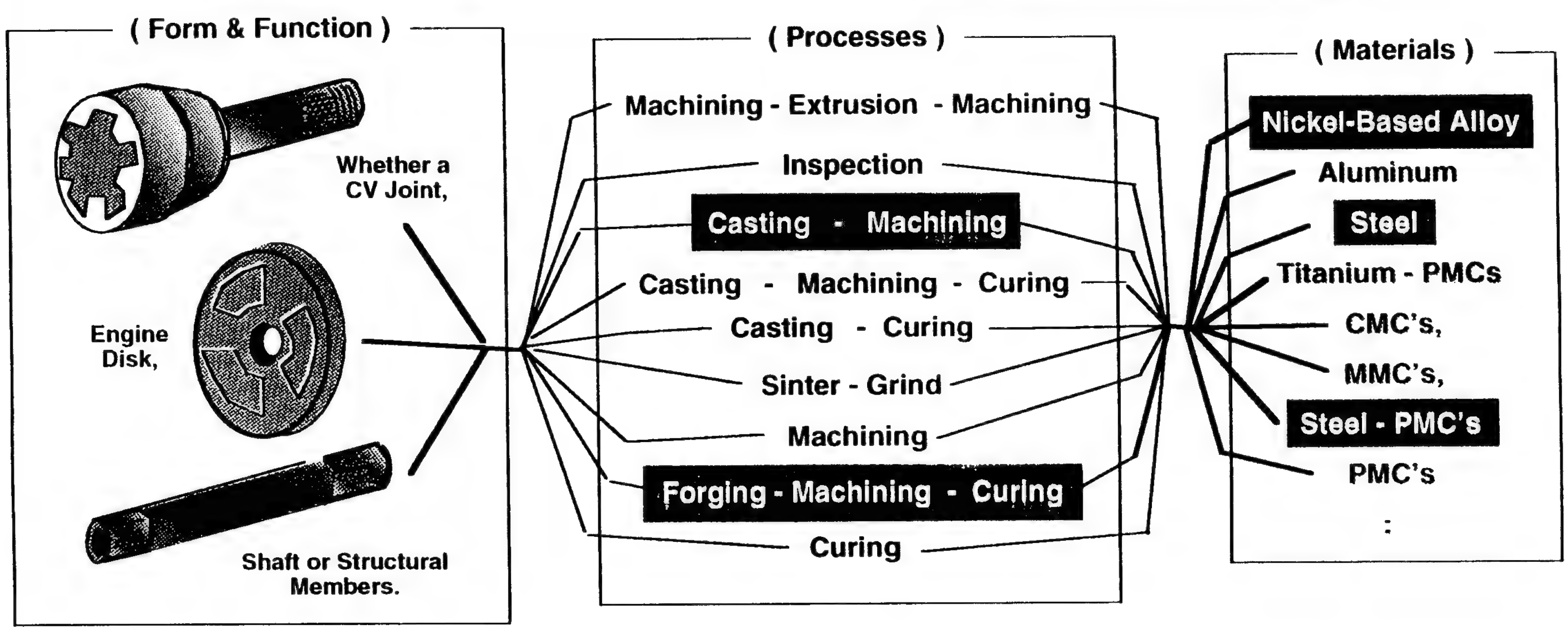
(Focus: Integrated Form & Function <--> Material & Process Design)

Competing Multiple and/orAlternative

(Casting, Forging-Extrusion, Machining, Sintering and Inspection)

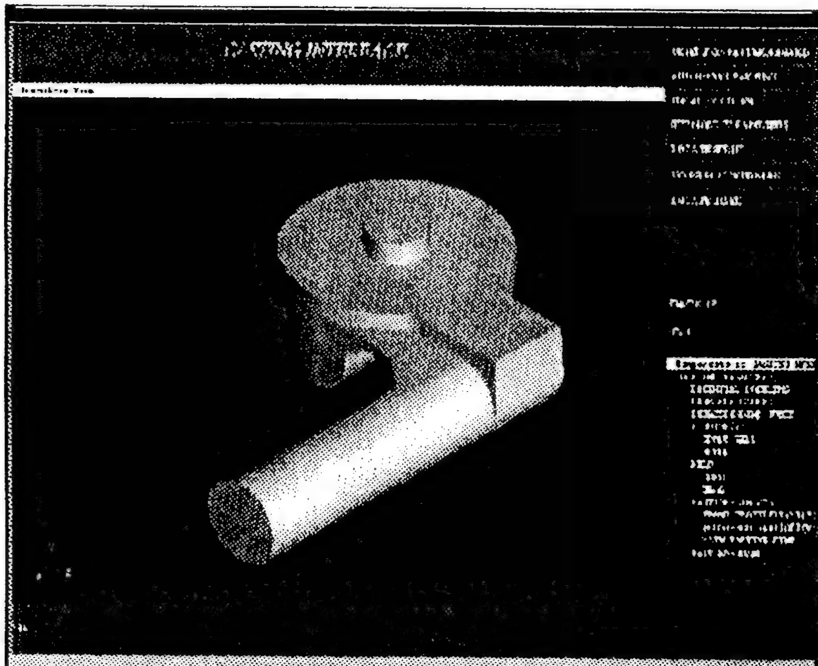
Material & Processes

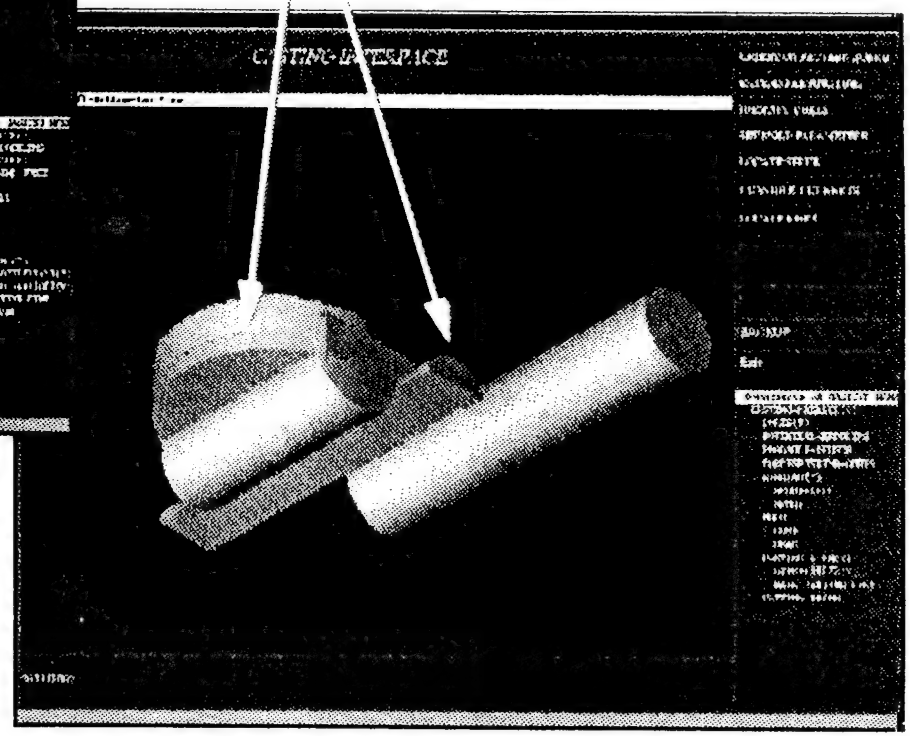
(Steel, Aluminum, Nickel-based & Titanium Alloys & Fiber Reinforced Composites)



RFTS: Optimized Recursive Design for

Foundry Tooling:
Coring, Draft, & Rigging





Air Force Invention
 # 20958

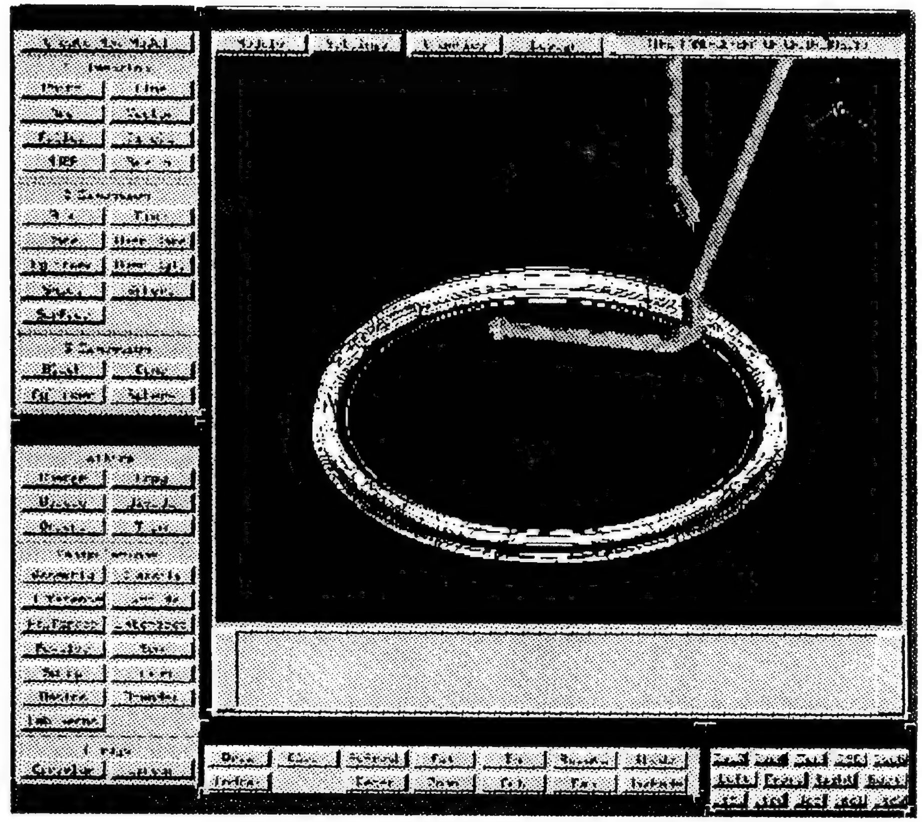
8

Optimized *In-situ* Inspection Probe Path Planning

Automated Path Sequencing with Collision Detection and Avoidance

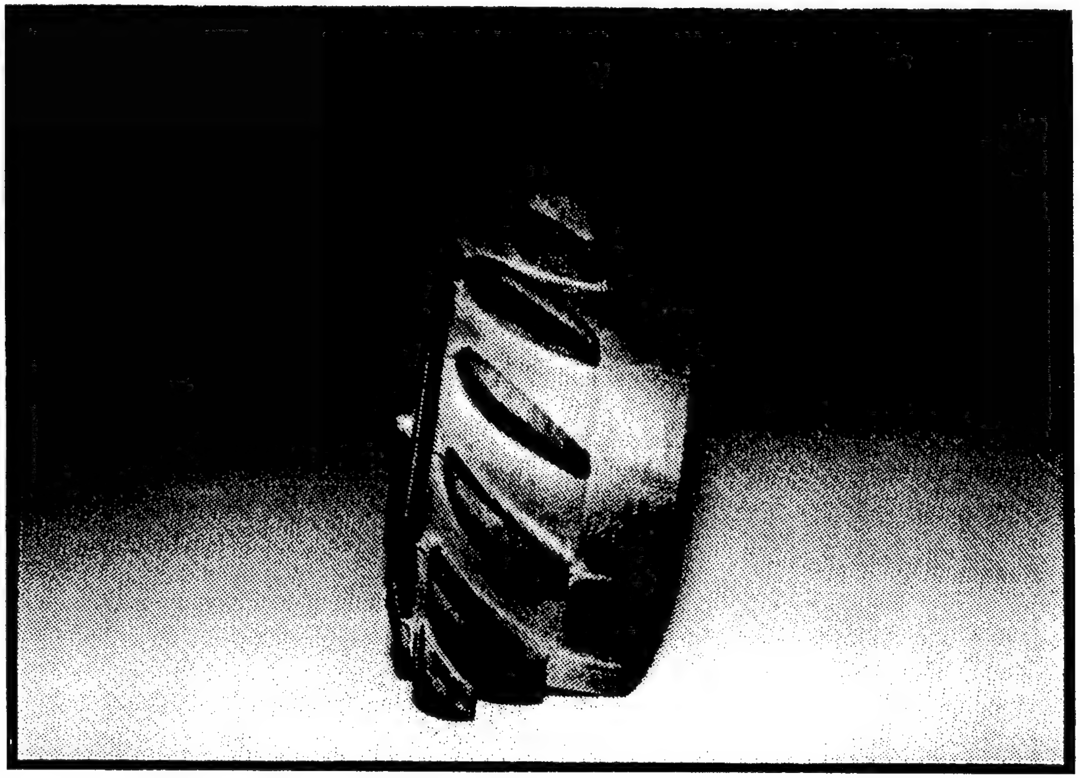
using

'Dupin Cyclide' or other Piping Algorithm and GA or other Discrete Optimization Method



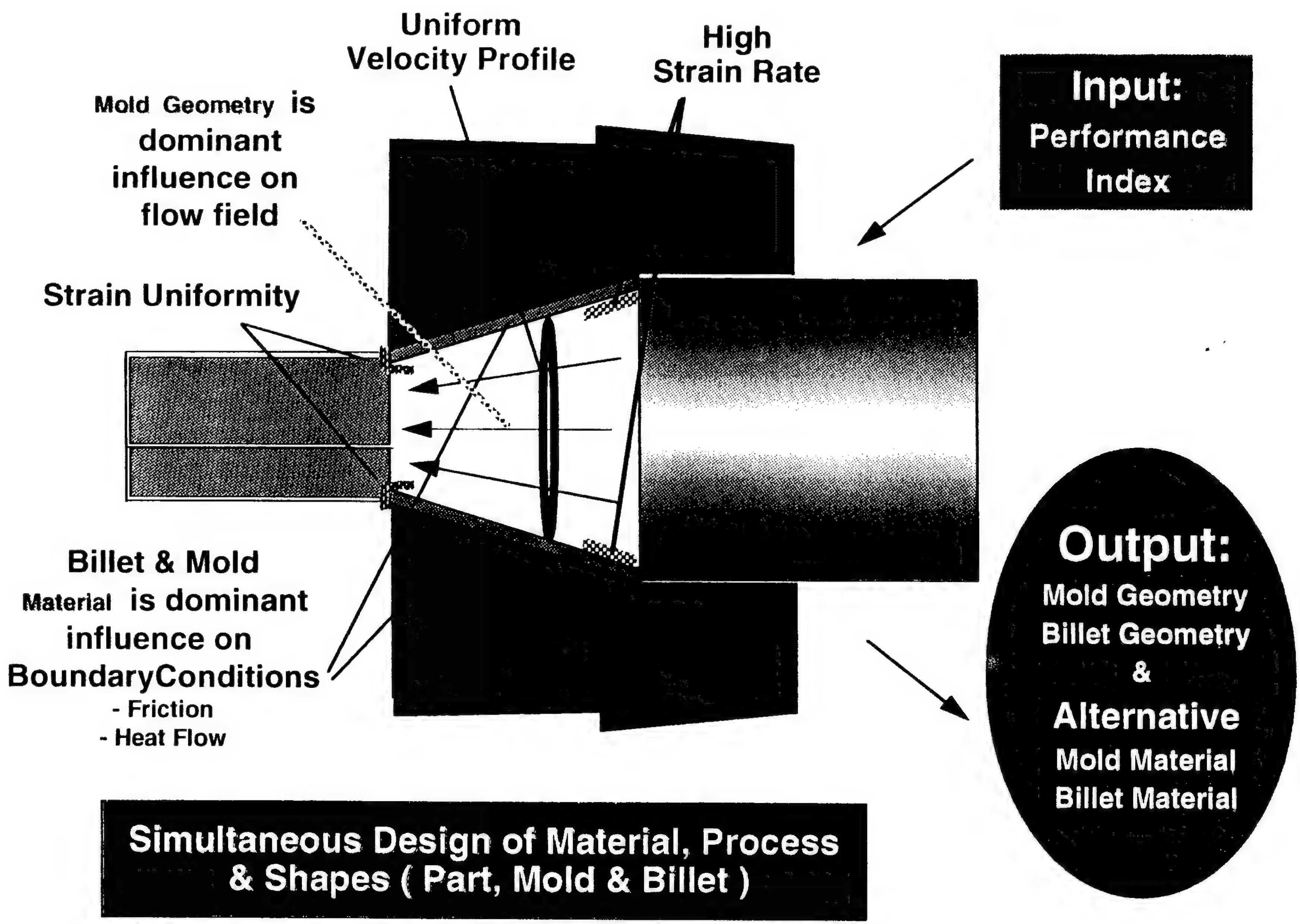
Materials Process Design

FORGED TiAl INTEGRATED BLADE AND ROTOR

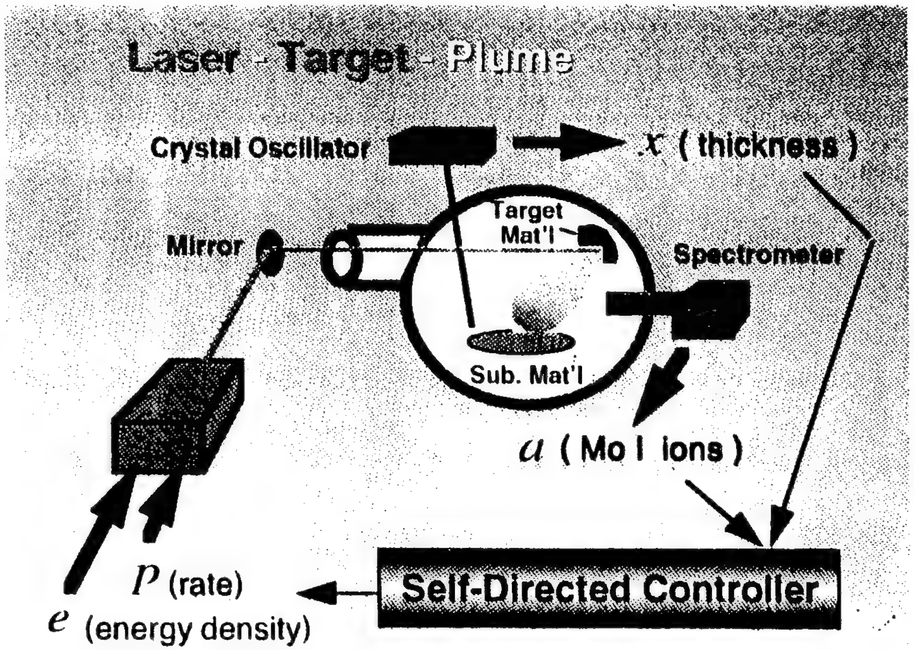


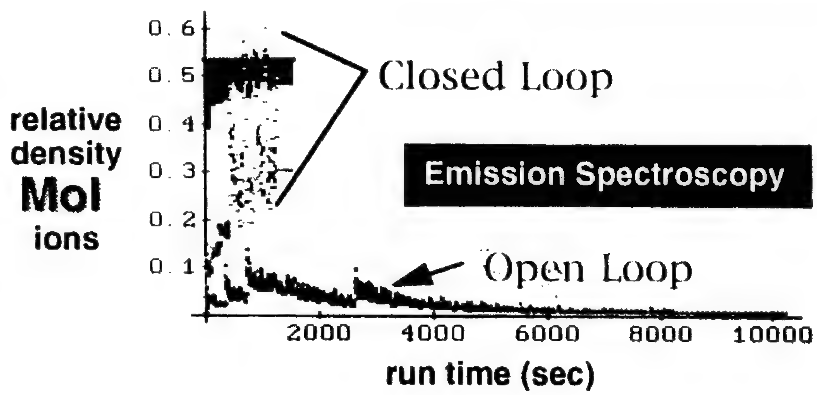
"State-Space" Process Design And Control Optimization Methodology Verification

lai95069/rp/1b-1


Uniform Velocity Profile
High Strain Rate
Input:
Performance Index
Mold Geometry is dominant influence on flow field
Strain Uniformity
Billet & Mold Material is dominant influence on BoundaryConditions
- Friction
- Heat Flow
Output:
Mold Geometry
Billet Geometry
&
Alternative
Mold Material
Billet Material
Simultaneous Design of Material, Process & Shapes (Part, Mold & Billet)

PLD *In situ* 'Remodeling'



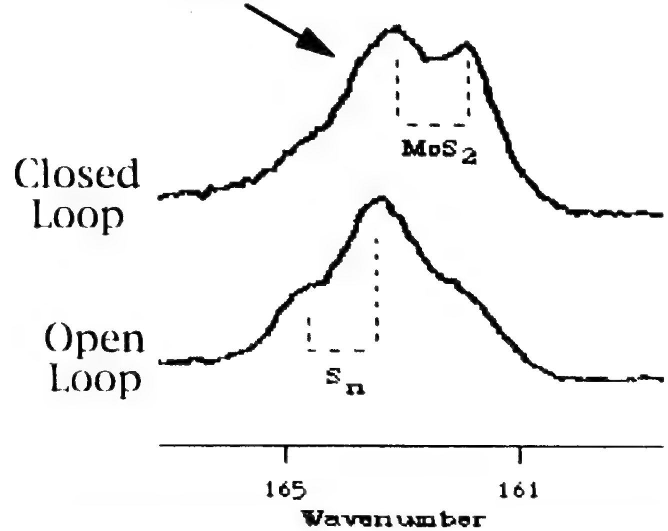


Emission Spectroscopy

- **SPEED**: 10.8X faster growth
  (0.027 Å/sec vs 0.295 Å/sec)

100% more material in 1/6 th the time enabling thicker films before 'coning' limits process

- **QUALITY**: Control of the material composition of the final film.



X-ray Photon Spectroscopy Results

Materials Process Design
(In-House Research)

1985 - 1989 SDC of Polymer Curing - 'Qualitative Process Automation'

Autoclave Curing of Polymer, Reinforced Composites - the knowledge base of rules used to parse sensed data, compare past with present material and process states, and associate mapping with predicted next state to in-situ construct a cure cycle. The research contributions were: 1) material-process self-direction, 2) adaptation to varying material and process conditions, 3) 70% reduction in cure times.

1990 - 1996 SDC of Pulsed Laser Deposition & Molecular Beam Epitaxy

Deposition of Tribological Thin-films - the knowledge base consisted of 3 rules to parse and detect film thickness, morphology and composition and invoke a non-linear regression model to in-situ remodel the process to determine new settings' of laser energy and pulse rate. Research contributions expected in: 1) material-process self-direction & discovery, 2) precise film property repeatability, 3) new materials.

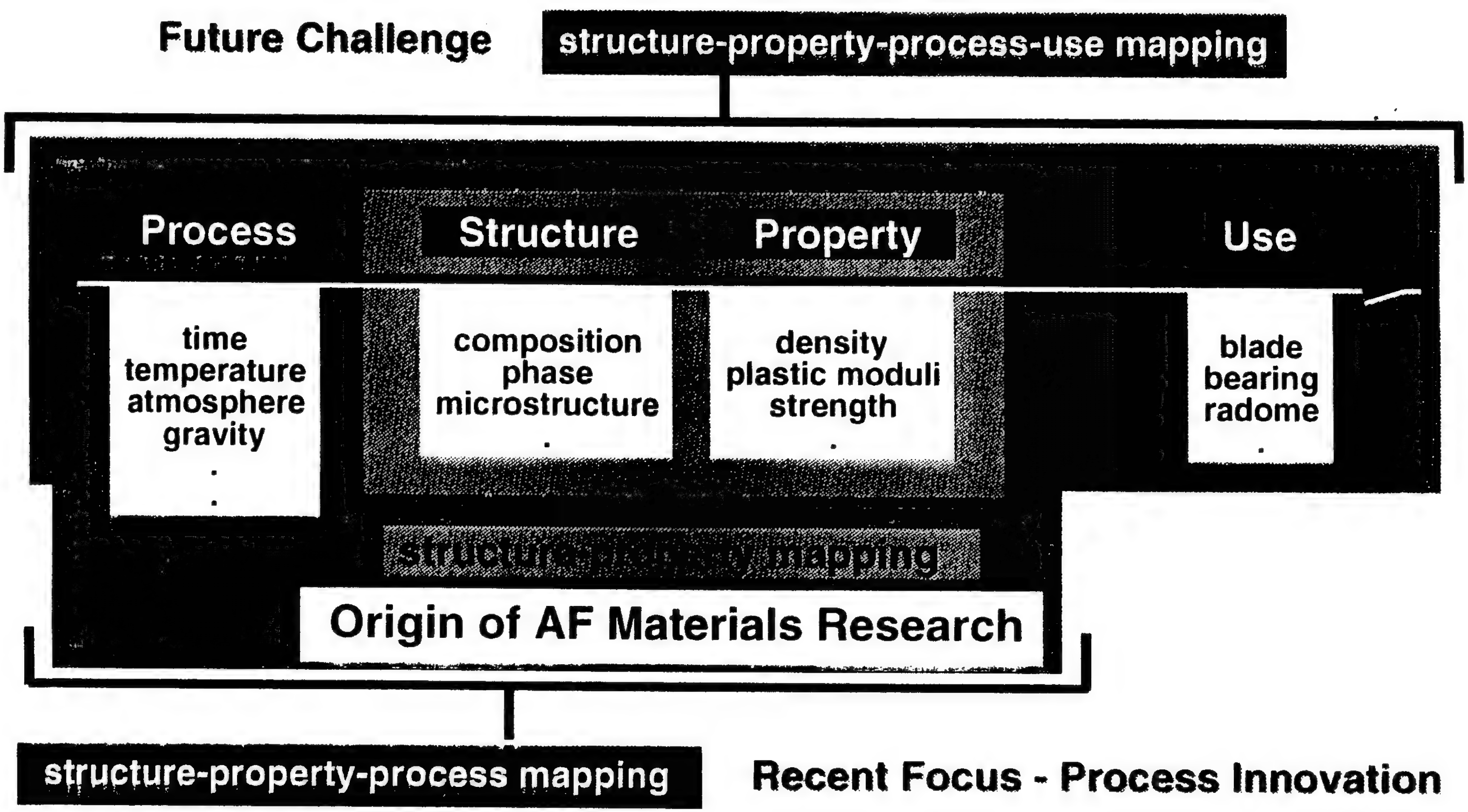
1995 - 1997 SDC of Chemical Vapor Deposition

Deposition of Interface Films - the knowledge base will include a neural net controller with a rule-based supervisor to parse and detect the film thickness, morphology and composition via in-situ characterization of the film and reactants. Contributions are expected in the area of: 1) material-process self-direction & discovery, 2) ability to adapt to varying material and process conditions, 3) precise molecular control of interface film to enable production of CMC's of desired strength-toughness.

1995 - 2000 Automated Materials Research & Process Discovery

Self-Directed Control/Process Discovery for Thin-film Processing - using FLN technology to automate non-linear regression and address: 1) error characterization, 2) in-situ adaptation, and 3) problem characterization/discovery. The essence of these contributions will be to automate materials research and augment researchers in improving in-house research productivity and affordability of new Materials & Processes.

Materials Process Design: Transition from
The First to the Next 75 Years



Future Directions

Air Force Wright Laboratory
Materials Directorate

? DoD Aerospace Materials Laboratory ?

? National Aerospace Materials Laboratory ?

FROM

Function

Structural <- - > E-O Materials

&

Morphology/Chemistry

Bulk <- -> Thin Films

TO

Functionally Gradient (Monolithic, Composites & Films)

Lightweight, High Temperature Materials

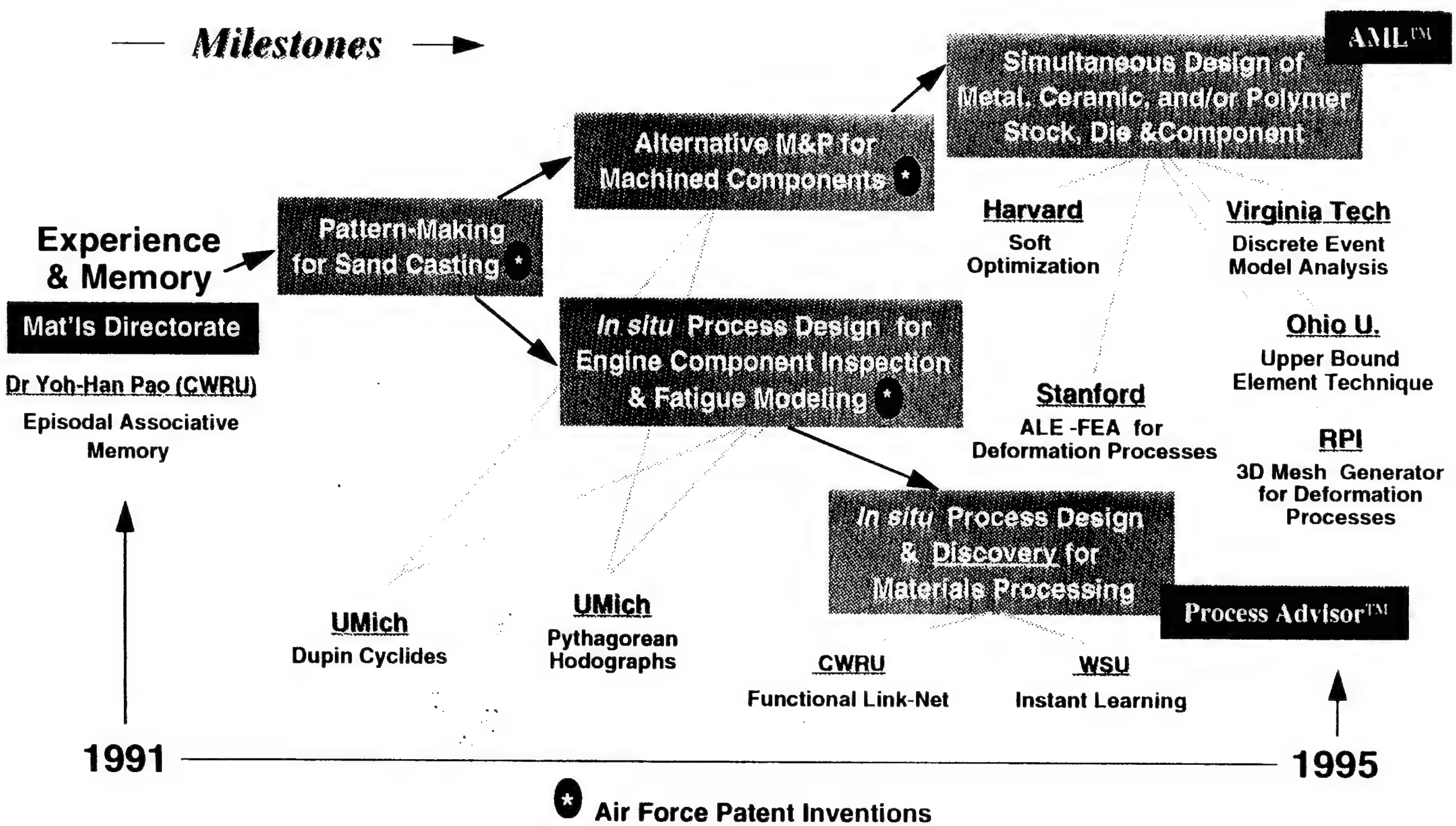
&

Controlled Composition & Morphology

Integrated (2D& 3D) Nano - Macro Scale

Electronic Prototyping

Computational Methods for Design



16

Discrete Event Design & Analysis

May 23:

8:30 Welcome
Dean Brandeberry, Wright State University

8:45 Opening Remarks: Electronic Prototyping
S. LeClair, Wright Laboratory

9:00 Soft Optimization of Hard Problems
Y.C. Ho, Harvard University

9:45 - 10:00 Break

10:00 Process Model Development for Use with Discrete Event System Techniques
J. Gunasekera, Wright Laboratory Visiting Scientist

10:45 Analyzing Discrete Event Simulation Models of Complex Manufacturing Systems: A Computational Complexity Approach
S. H. Jacobson & Capt A.W. Johnson, Virginia Tech

Soft Optimization of Hard Problems

by Y.C. Ho
Harvard University

Abstract

It can be argued that optimization in the general sense of making things BETTER is the principal driver behind all of prescriptive scientific and engineering endeavor, be it operations research, control theory, and engineering design. It is also true that the real world is full of complex decision and optimization problems that we cannot solve. While the literature on optimization and decision making is huge, much of the concrete analytical results are associated with what may be called Real-Variable-Based methods. The idea of successive approximation to an optimum (say, minimum) by sequential improvements based on local information is often captured by the metaphor of "skiing downhill in a fog". The concepts of gradient (slope), curvature (valley), and trajectories of steepest descent (fall line) all require the notion of derivatives and are based on the existence of a more or less smooth response surface. There exist various first and second order algorithms of feasible directions for the iterative determination of the optimum (minimum) of an arbitrary multi-dimensional response or performance surface. Considerable number of major success stories exist in this genre including the Nobel prize winning work on linear programming.

On the other hand, we submit that the reason many real world optimization problems remain unsolved is partly due to the changing nature of the problem domain in recent years which makes calculus or Real-Variable-Based methods less applicable, e.g., , a large number of human-made system problems, such as manufacturing automation, communication networks, computer performances, and/or general resource allocation problems, involve combinatorics rather than real analysis, symbols rather than variables, discrete instead of continuous choices, and synthesizing a configuration rather than proportioning the design parameters. Such problems are often characterized by the lack of structure, presence of large uncertainties, and enormously large search space. Optimization for such problems seem to call for a general search of the performance terrain or response surface as opposed to the "skiing downhill in a fog" metaphor of Real-Variable-Based performance optimization (footnote #1) . Arguments

can also be made on the technological front. Sequential algorithms were often dictated as a result of the limited memory and centralized control of earlier generations of computers. With the advent of modern general purpose parallel computing and essentially unlimited size of virtual memory, distributed and parallel procedures or a network of machines can work hand-in-glove with Search-Based methods of performance evaluation. It is one of the thesis of this talk to argue for such a complementary approach to optimization.

If we accept the need for Search-Based methods as a complement to the more established Real-Variable-Based analytical techniques, then we can next argue that to quickly narrow the search for optimum performance to a "good enough" subset in the design universe is more important than to estimate accurately the values of the system performance during the initial stages of the process of optimization. We should compare order first and estimate value second, i.e., ordinal optimization comes before cardinal optimization. Colloquial expressions, such as "ballpark estimate", "80/20 solution" and "forest vs. trees", state the same sentiment. Furthermore, we shall argue that our preoccupation with the "best" may be only an ideal that will always be unattainable or not cost effective. Real world solutions to real world problems will involve compromise for the "good enough" .

In summary, the purpose of this talk is to establish the distinct advantages of the softer approach of ordinal optimization for the Search-Based type of problems. More specifically, the presentation will address general properties, show the many orders of magnitude improvement in computational efficiency that is possible under this mind set, and finally advocate the use of Soft Optimization as a complement to the more traditional cardinal approach to engineering design. Example applications such as rare event probability in systems, ATM switches in communication, buffer allocation in manufacturing, and control system design will be highlighted to substantiate the case (footnote #2). Relationships to other subjects such as fuzzy sets, genetic algorithms, and rank statistics will also be mentioned.

Footnotes

#1 We hasten to add that we fully realize the distinction we make here is not absolutely black and white. A continuum of problem types exist. Similarly, there is a spectrum of the nature of optimization variables or search space ranging from continuous to integer to discrete to combinatorial to symbolic.

#2 Joint work with Ph.D. students, Mei Deng, M. Larson, E.T.W. Lau, L.H. Lee, D.W.G. Li, N. Patsis

Soft Optimization of Hard Problems

by
Yu-Chi Ho
Harvard university

May 23-24 1995
WPAFB, Ohio

20

Characteristics of "HARD" Problems

- *Lack of Structure*: combinatorial, discrete, and symbolic variables,
- *Uncertainties*: inherent part of the problem; requires time consuming averaging; complex and ill-defined,
- *Huge Search Space*: combinatorial explosion; not easily parametrized

Simulation only tool to optimize or validate ad hoc designs

21

Examples of "HARD" Problems

- Design of scheduling rule in ATM switches
- Buffer allocation in production line
- Feedback control law in nonlinear control problems
- Engineering design of complex and not completely understood manufacturing process

•
•
•

23

Performance Evaluation

- *Modeling & Programming* 40%
- *Analysis and Validation* 10%
- *Running the Simulation* 25%
- *Optimizing the Design* 25%

According to a Winter Simulation Council study

24

Ordinal Optimization Approach to Hard Pbs.

- facilitate initial "narrowing down" of choices
- quantify heuristics and marginal cost effectiveness
- complementary to ANN, FL, GA and other AI tools
- *potential* resolution of the combinatorial explosion problem.

Idea of Ordinal Optimization:

The first goal of optimization is to get good enough designs— ordinal

Softer question yields more interesting answers

Then worry about estimating the performance value of the best design— cardinal

The ideal may not be cost-effective

—Y.C. Ho 1993—

Trade Off Analysis (contd)

$$\boxed{P=1-(1-n\%)^N \; ; \; CFI \sim 1/(N)^{1/2}}$$

- High Probability (0.9) Certainty (0.9999)

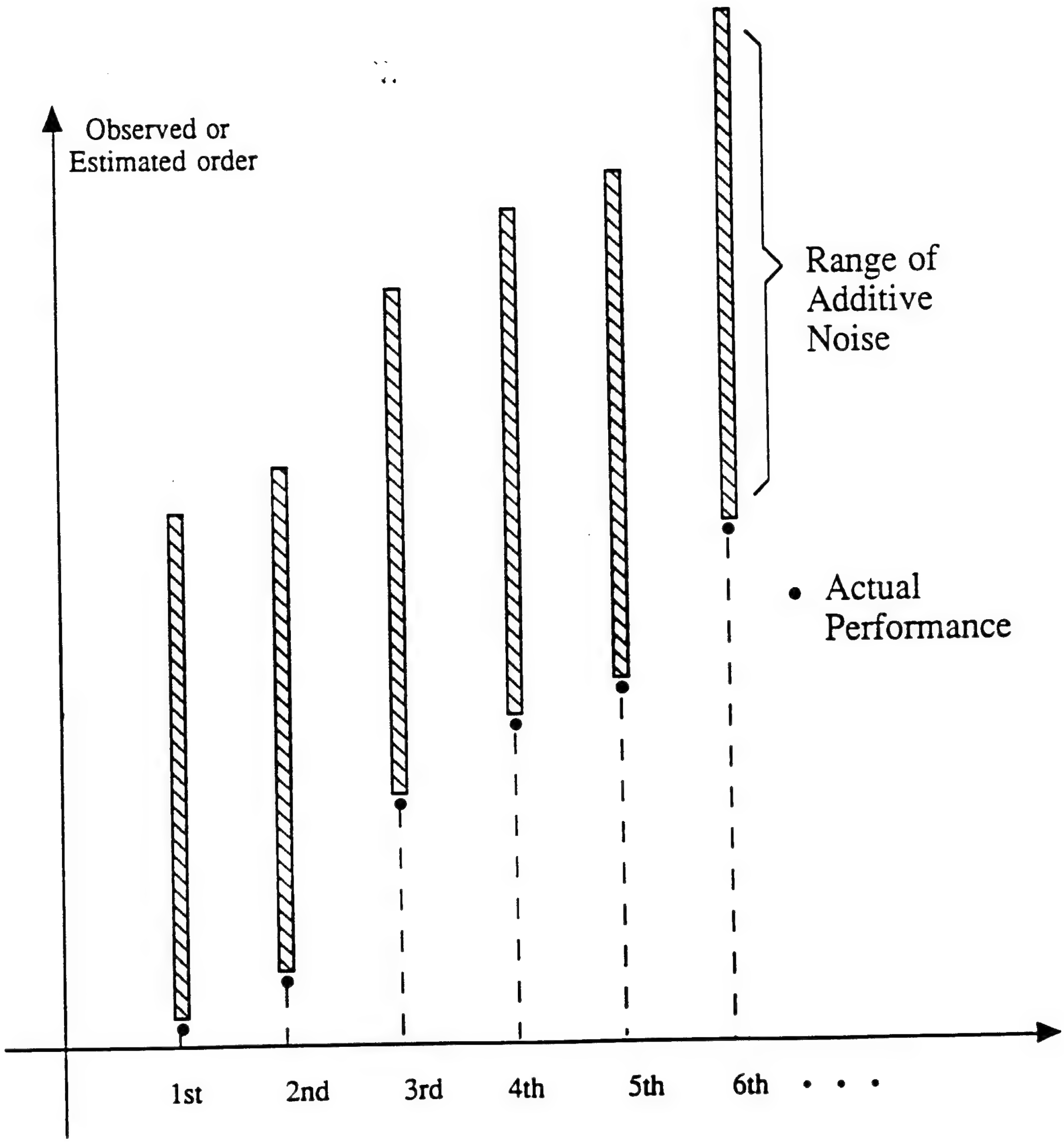
==> *4 times (or 10^6) more sampling cost*

- Good Enough (Within top-5%) The Best (Within top-0.005%)

==> *1000 times more sampling cost*

- Cost are multiplicative ==> *4000 times* *10^{12} times* if CFI criterion is used

Je



Fundamentals of Ordinal Optimization (contd.)

—Y.C. Ho 1993—

ASKING SOFTER QUESTIONS

- The "Good Enough" group, G,

  e.g., the top ten designs in the set of all possible designs

- The "Selected" group, S,

  e.g., the estimated top ten design selected by some rule

$$\boxed{\text{Alignment} = G \cap S}$$

29

In Traditional Optimization,

G = {best} & S = {estimated best}

metaphor: *hiting one bullet with another*

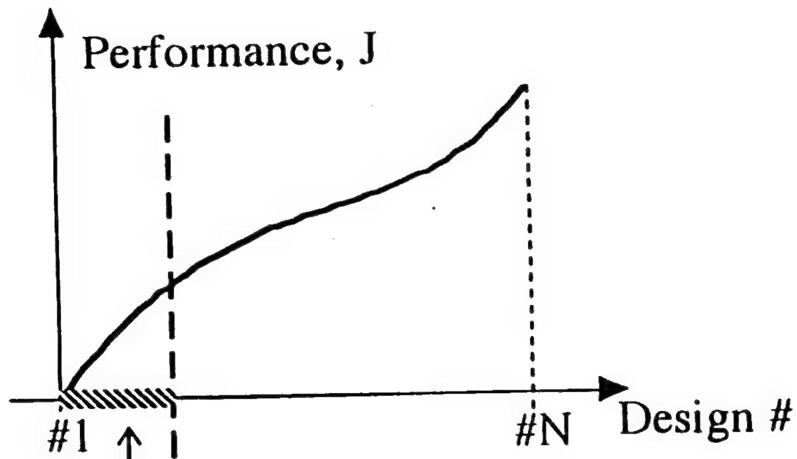
In Ordinal Optimization

G = {Good Enough subset} &

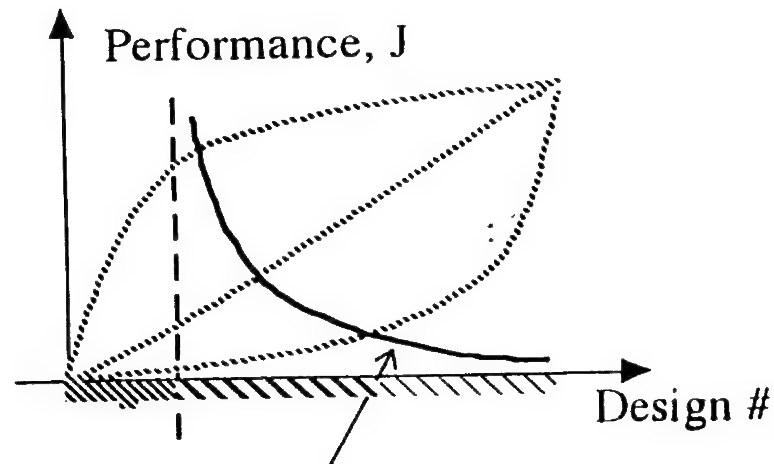
S = { Selected Subset}

metaphor: *hitting a truck with a shot gun*

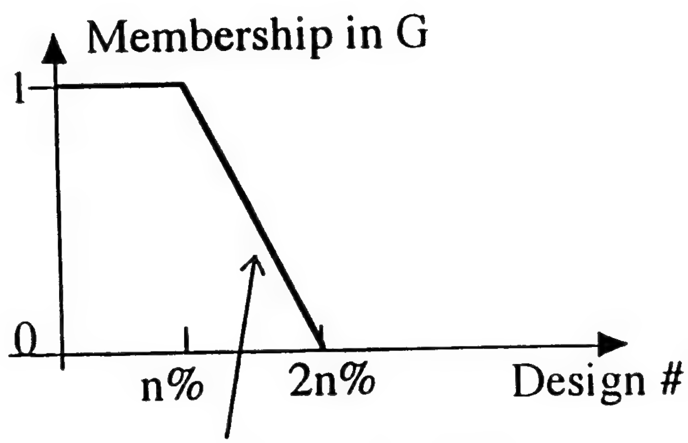
Definition of the "Good Enough" Subset, G



Simple def. for G = top n%



A J-dependent def. of G



Fuzzified def. of G

And many other reasonable definitions for G,

For exposition purposes, we'll use the simple definition of TOP-g DESIGNS

Rules for Selecting the Subset , S

- Blindly Pick (BP)

  easily done; no knowledge required

- Horse Race (HR) : pick the observed top-n

  requires observing all N alternatives

- Tournament Elimination (TE):

  needs only pair-wise comparison

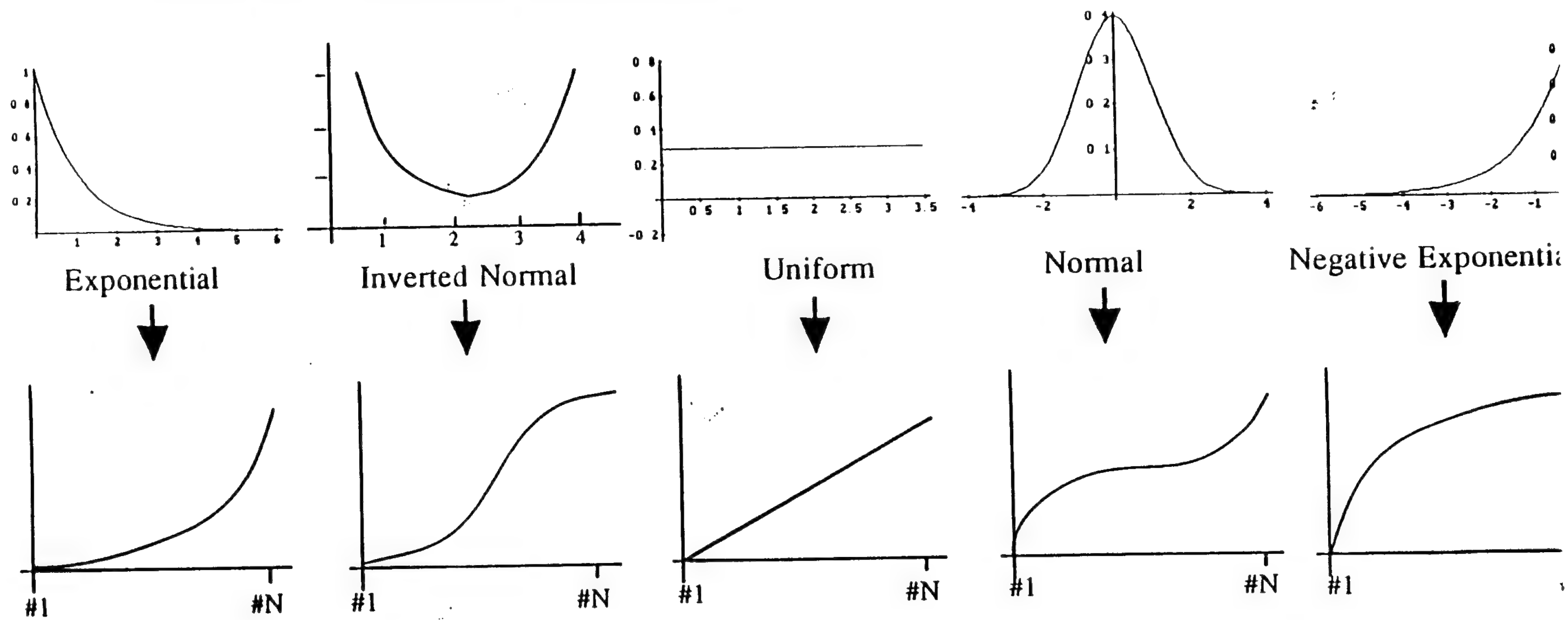
- Round Robin Play (RR):

  most reliable, least efficient

- Other Heuristic Rules:

  how to quantify?

Generic Performance Distributions and Their Related Ordered Perfromance Curves



Near the origin (optimum) there are only three kinds of slopes for the OPC: Steep, Flat, Neutral

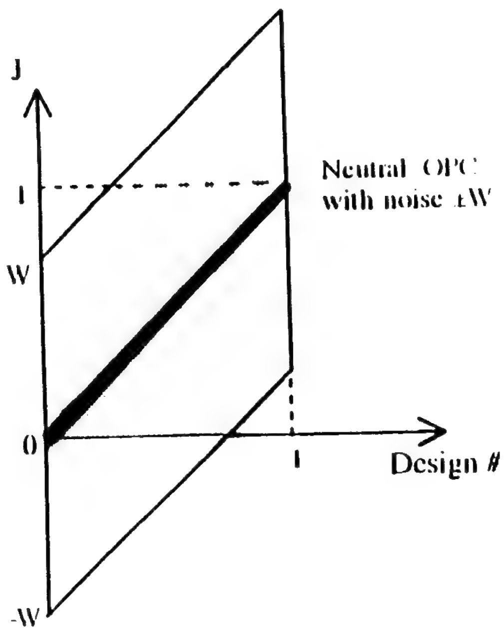
Universal Alignment Probability Curves
(contd)

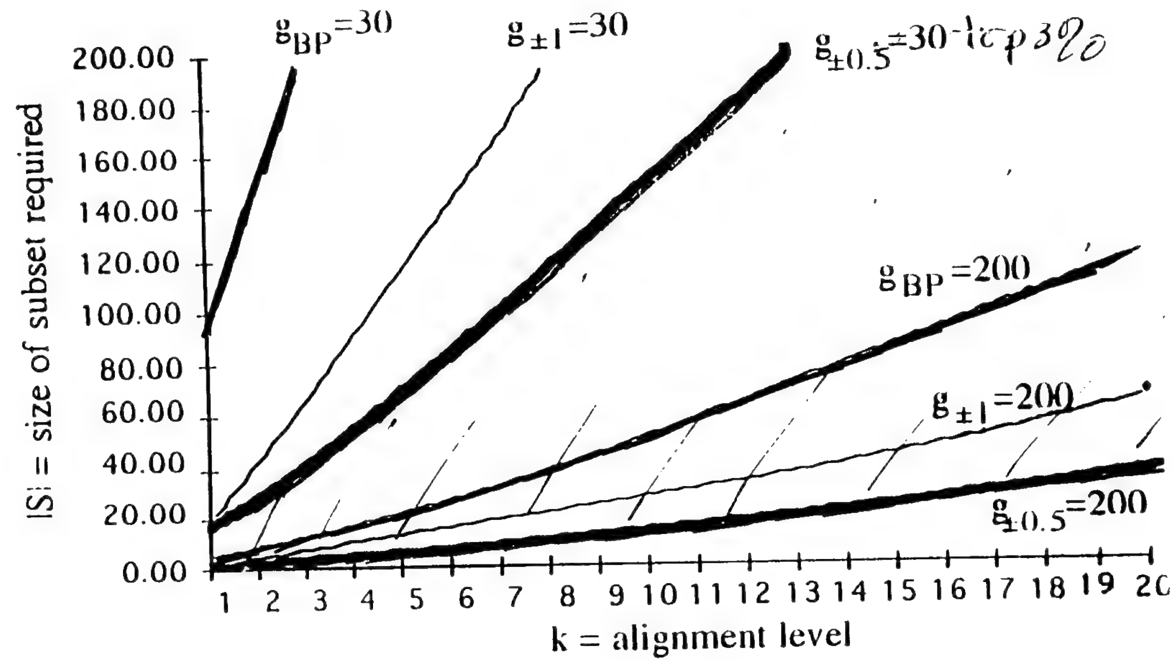
We have calculated the function $P(|G \cap S| \geq k; g, s, \sigma^2, C, N)$ extensively. In particular, we set $N=1000$, $P=0.95$ and plotted the "s" required for different values of g, k, σ^2, and C.

In other words, we answer the question:

"If we roughly evaluate N performance alternatives, how far can we narrow down the choices, i.e., from N->s, if we wish to be assured that at least k truly good alternatives are in the reduced set, S?"

Universal Alignment Probability Curves
(contd)





C=uniform distributed, J range (0,1) ;
W = noise range $\pm\infty$, ±1, ±0.5; N=1000

Example Applications

- **to Manufacturing Automation:** *Ho, Y.C., Sreenivas, R., Vakili, P., "Ordinal Optimization of Discrete Event Dynamic Systems", J. of DEDS 2(2), 61-88, (1992).*

- **to Communication Networks:** *Patsis, N., Chen, C.H., & Larson, M., "Parallel Simulation of DEDS", Proceeding of Optimization Days, (1993) to appear in IEEE Trans. on Control Technology (1995);*
  Wieseltheir, J.E., Barnhart, C.M., and Ephremides, A., "Ordinal Optimization of Admission control in Wireless Multihop Voice/Data Nework via Standard Clock Simulation", to appear in J.DEDS, 1995

- **to Reliability (Rare Event Probability):** *Y. C. Ho and M. Larson, "Ordinal Optimization and Rare Event Probability Simulation" to appear in the J. Discrete Event Dynamic Systems, 1995*

• • • and others • • •

Using Simpler Surrogate Model to Predict Order in Complex Systems

- For M/M/1/K Queue: Buffer Overflow Probabilities (BOP) as a function of ρ, the traffic intensity are ordered equivalently for $K \geq 2$.
- Surrogate Model #1: For 3-node tandem queue: we assume BOP is similarly order equivalent for lower K values
- Surrogate Model #2 (lower K value plus rational approximation extrapolation, *Ref. W.B. Gong et al : 1994 CDC paper*)
- Surrogate Model #3: Using shorter simulation runs to predict steady state BOP.
- Surrogate Model #4: Combinations of #3 with #1 or #2

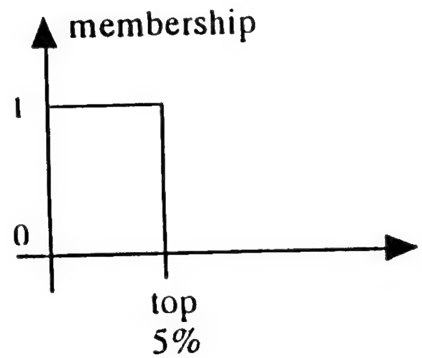
Relationship & Differences with Ranking Statistics

- Population size in billions rather than tens
- No cardinal notion
  - — not concerned with the distance of the best from the rest
- Different questions, related model
  - — Is the observed order the MLE of the true order?
  - — The Thurston-Mosteller-Daniel Model (additive noise)

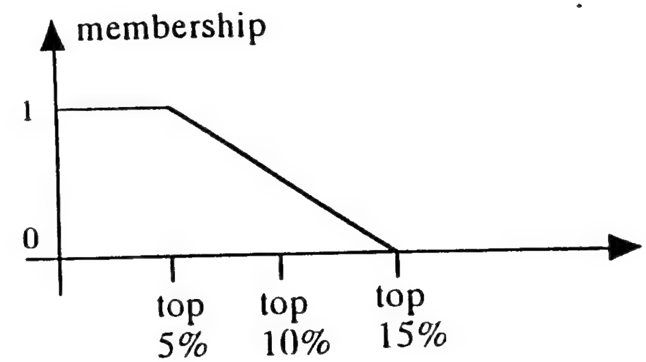
Example of Fuzzifying the Ordinal Approach

- Definition of "good Enough"

From

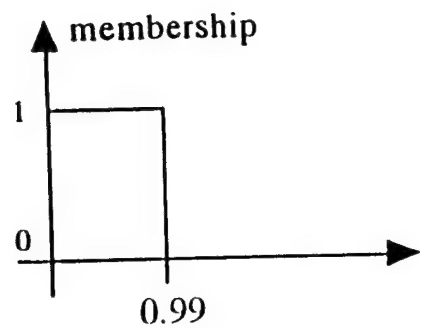


To

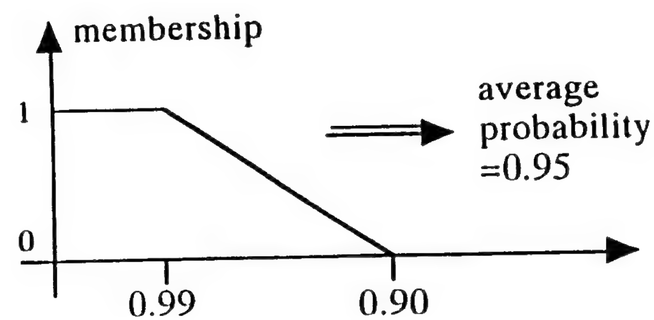


- Definition of High Probability

From



To



39

ORDER converges faster than **VALUE**

At least @ rate 1/t and sometimes exponentially fast

A rigorous proof:

Dai, L.Y. "*Convergence of Ordinal Comparison in the Simulation of Discrete Event Dynamic Systems*" Manuscript 5/95 Washington U. St. Louis.

Sampling & Search in Strategy Space

Problem: Find $u_t=\gamma(x_t)$ to minimize $J = E\{\lim_{t\to\infty} \frac{1}{T} \Sigma_t [x_t^2 + u_t^2]\}$ subject

to $x_{t+1} = -x_t+u_t +w_t$ where w_t is GWN(0,25)

— a simple Linear-Quadratic Pb.

Strategy Space, Γ: Quantize x to n values and u to m values

$\Longrightarrow |\Gamma| = n^m$ e.g., n=m=100 , $|\Gamma^{100}|=100^{100}$, n=5, $|\Gamma^5|=5^{10}$

— enormous search space.

Known Optimal Solution, γ^*: $u_t = 0.618x_t$; $J^*= 40.45$

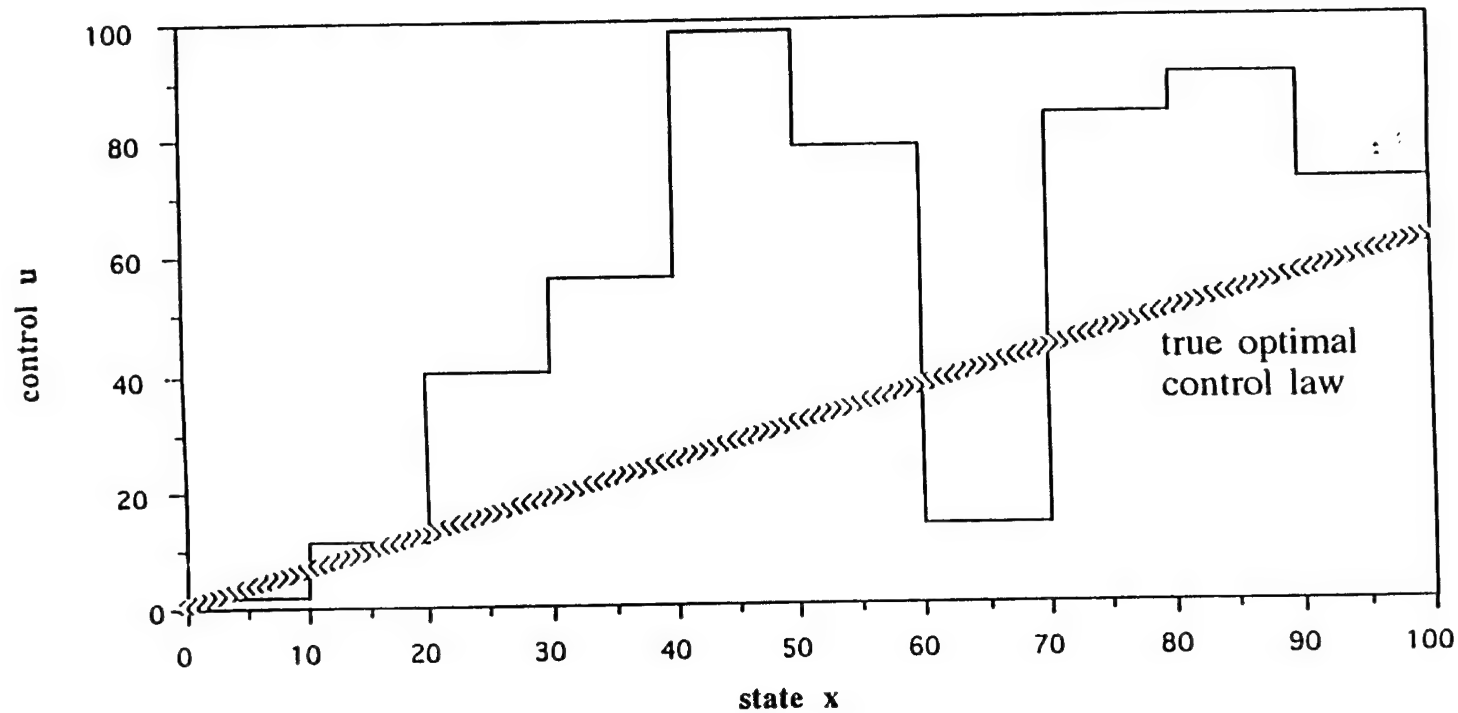
— optimal solution is linear

Plan:

SAMPLE 1000 γ'S UNIFORMLY IN Γ

40

Sampling in Γ^{10} Space



Best Control Law found => J=53.9

Ordinal Optimization Summary

OO is complementary to:

- **traditional optimization methods**
  - — initial search vs. end game
- **genetic algorithm and fuzzy logic**
  - — softening the goal vs. softening the definition of "optimum"
  - — makes function evaluation easier vs. to learn via function evaluation
- **parallel computation**
  - — permit use of simpler models, multiple versions of the model, cooperative search, etc.

Process Model Development for use with Discrete Event System Techniques

by Jay S. Gunasekera,
Ohio University and
Visiting Research Scientist
Wright Laboratory

Abstract

In the past decade, process analysis and modeling has enjoyed a resurgence owing largely to the continued growth in the performance-to-cost ratio of computational systems. Traditionally, process modeling has been limited to analysis of a single step in a typical material processing sequence, and process optimization has generally been carried out empirically by iteratively testing and improving designs for each step in the process.

The current investigation is aimed at developing process designs which will enable the analysis and optimization of a material processing sequence. Typically, such a sequence may involve several stages and alternative processing paths for a given component. It may also involve both Real-Variable and Search-Based (discrete event) models. Of principle research interest is that the material processing sequence be globally optimized to achieve improvements in affordability, producibility and performance of Air Force Systems.

Material process models are capable of representing and analyzing complex material flow problems (2D and 3D) and non-isothermal deformation processes with a high degree of accuracy. Material process models are generally non-linear and therefore, rely on numerical techniques for solutions. These models attempt to simulate the kinetics of material processing, and typically consist of inputs, outputs, constraints and a set of equations to be differentiated and/or integrated which describe the physics of the process. The model may be treated as a "Black Box" by the optimization algorithm. The entire manufacturing process for a "state-of-the-art" engine part (ex: Integrated Blade Rotor, IBR) consists of several individual discrete processes, such as casting, HIPing, forging, heat treating, machining etc. Each of these processes has specific inputs, outputs, equipment settings, constraints and workpiece information. The model

incorporates all of this information internally and generates variables for a set of specific inputs.

There are several methods available for the analysis and modeling of material deformation processes. They include lumped models (i.e., no distribution of variables), slab models (i.e., geometric simplifications), upper bound models (UBET) models, finite difference models and finally, finite element models. The former models are very fast in terms of computer execution times but provide less detailed information, whereas the latter models are very accurate and are capable of predicting the variables in 2D or 3D, but are very time consuming. The fundamental approach in the discrete event simulation model is to use models which are fast during the early phase of optimization and progressively increase the accuracy of the models when the optimum is reached.

This paper will demonstrate the development of multiple levels of models for forging and heat treatment processes. The models are being incorporated into an overall discrete event simulation model and will be used to determine the global optimum for the entire material processing sequence which will ultimately involve billet preparation and finish part machining operations.

• The overall goal of this research is to establish a systematic methodology for optimizing complex manufacturing processes with respect to affordability and quality.

• This research focuses on the development of analytical and numerical models for a wide range of material processes which are important to production of Air Force systems.

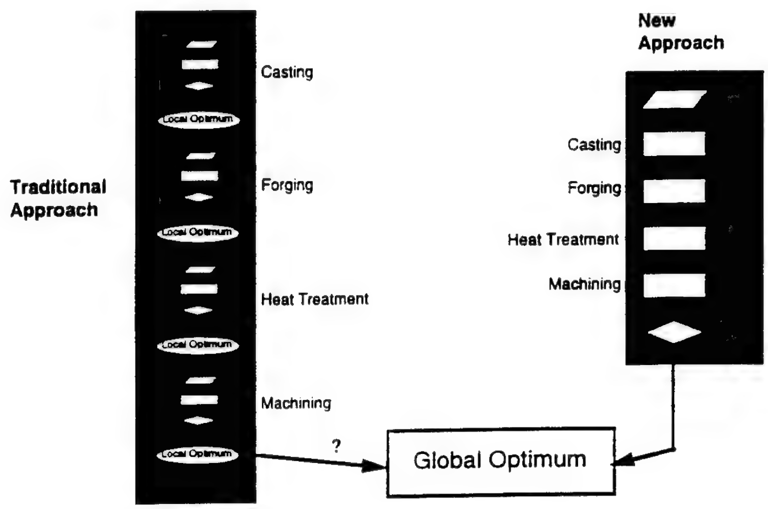
Photo of an extruded-blade IBR during fabrication.

- Components for Air Force systemstypically involve advanced materials and intricate geometries with specified microstructures and mechanical properties.

- Photo of integral blade and rotor component.

- Drivers for System Performance
  - Ability to effectively produce integral structure components
  - Ability to effectively make shapes from advanced materials
  - Ability to achieve Microstructure & Property control in "Large Sections"

COMPARISON OF OPTIMIZATION APPROACHES

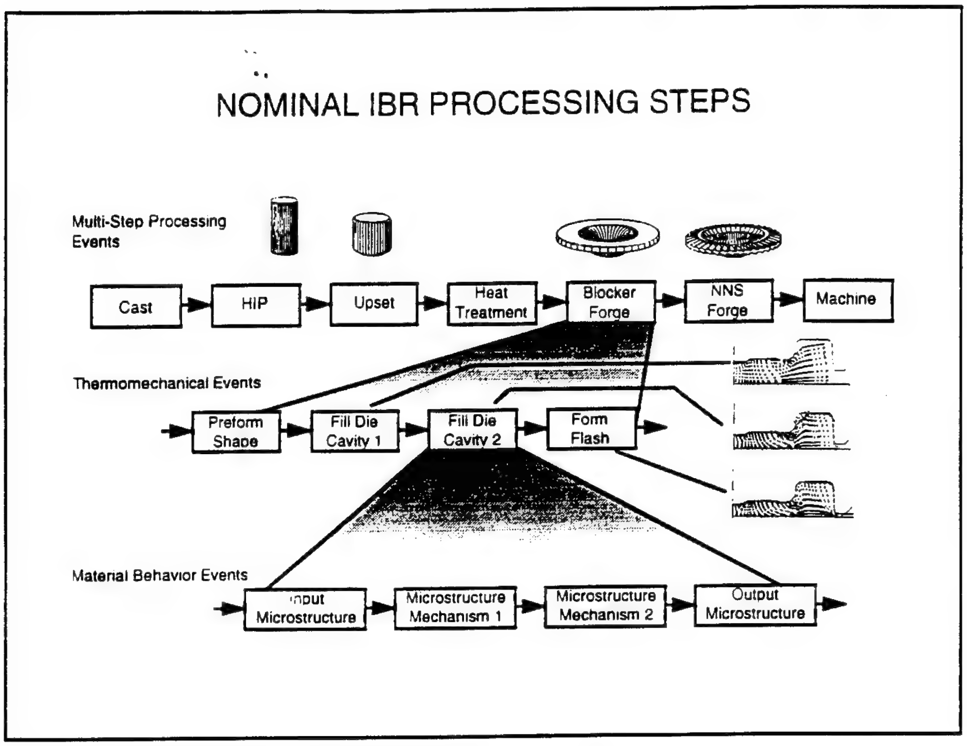


Traditional Approach:

- •Start with a feasible process as baseline.
- •Optimize each step in the process individually with knowledge of the processing operation.
- •Perform some trade-offs between the processing steps if of obvious advantage.
- •Hope/Assume/Pray the result is a global optimum for the process.

New Approach:

- •Treat each step as a single event in a series required to produce the part.
- •Perform optimization of the entire series by explicitly trading off the individual steps.



Typical IBR forming operations with associated shapes.

•Simple IBR shape from many processing steps.

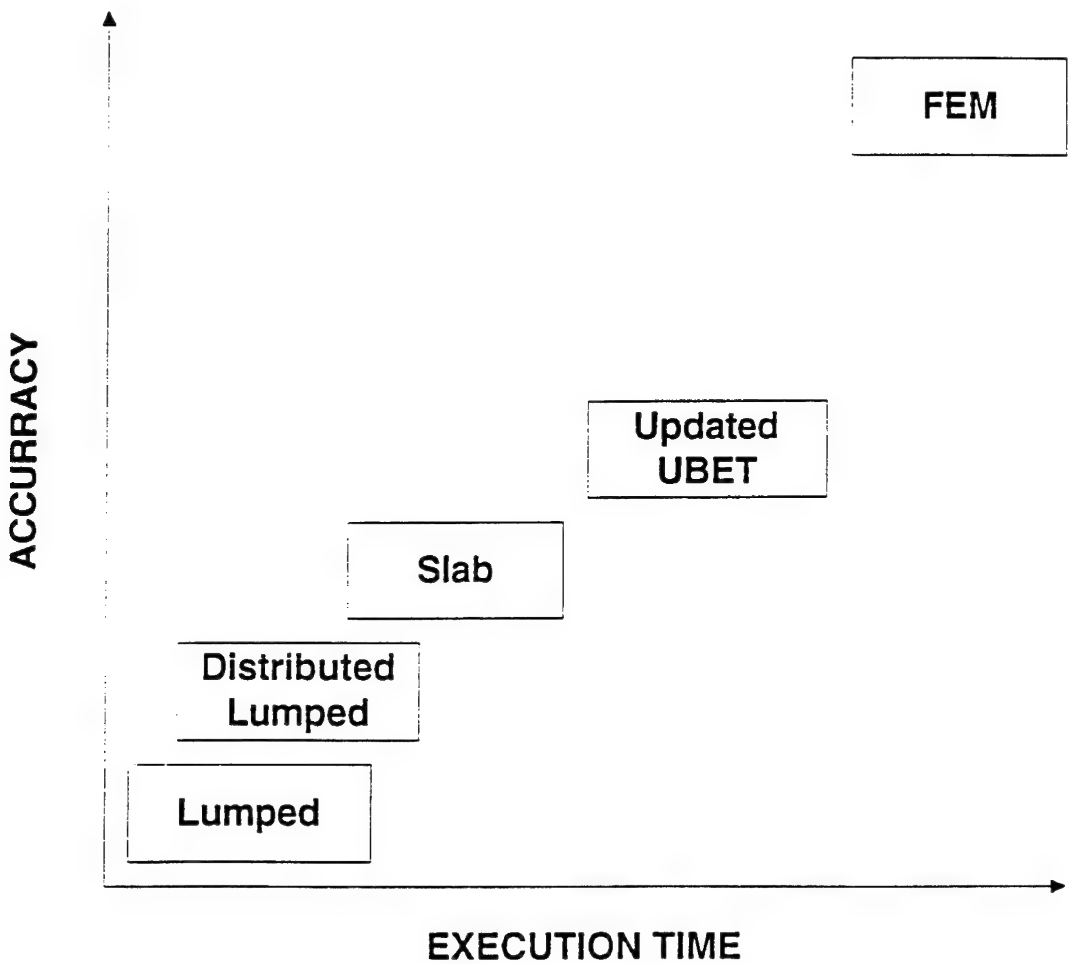
•Each step can be broken down into events.

•Each event can be further broken down into multiple material behavior events that determine flow (pattern filling) and microstructure.

•On this level...the entire process is a series of events that occur to produce the final part.

•The goal is to optimize the process.

Accuracy Vs. Execution Time of Various Models



Modeling Methods for Deformation Processes

- Slab
  - No Motion
  - Predicts Die Forces
  - Cannot Predict Fill
- Updated UBET
  - 2 or 3 Dimensional Models
  - Capable of Rough Property Distribution Predictions
  - Capable of Simulating Die Movement and Predicting Geometry and Fill
- FEM

PLASTICITY
Governing Equations

Equilibrium

$$\frac{\partial \sigma_{ij}}{\partial x_i} = 0$$

Yield Criterion

$$\bar{\sigma} = \sqrt{\frac{3}{2}}\,(\sigma'_{ij} \quad \sigma'_{ij})^{\frac{1}{2}} = f(\dot{\epsilon},\epsilon,\tau)$$

Constitutive Equations

$$\dot{\epsilon}_{ij} = \frac{3}{2}\frac{\dot{\epsilon}}{\bar{\sigma}}\sigma'_{ij}$$

$$\bar{\epsilon} = \sqrt{\frac{2}{3}}\,(\dot{\epsilon}'_{ij} \quad \dot{\epsilon}'_{ij})^{\frac{1}{2}}$$

Compatibility Conditions

$$\dot{\epsilon}_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

MODIFIED SLAB ANALYSIS
Governing Equations

Vertical Equilibrium

$$-\sigma_z = P_l\,(1-\tan\,\alpha)$$

$$= P_u\,(1+\tan\,\beta)$$

Horizontal Equilibrium

$$h\frac{d\sigma_r}{dr} + \frac{h}{r}\sigma_r + \sigma_r\frac{dh}{dr} + P_u\,\tan\,\beta - \tau_u - P_l\,\tan\,\alpha - \tau_l = 0$$

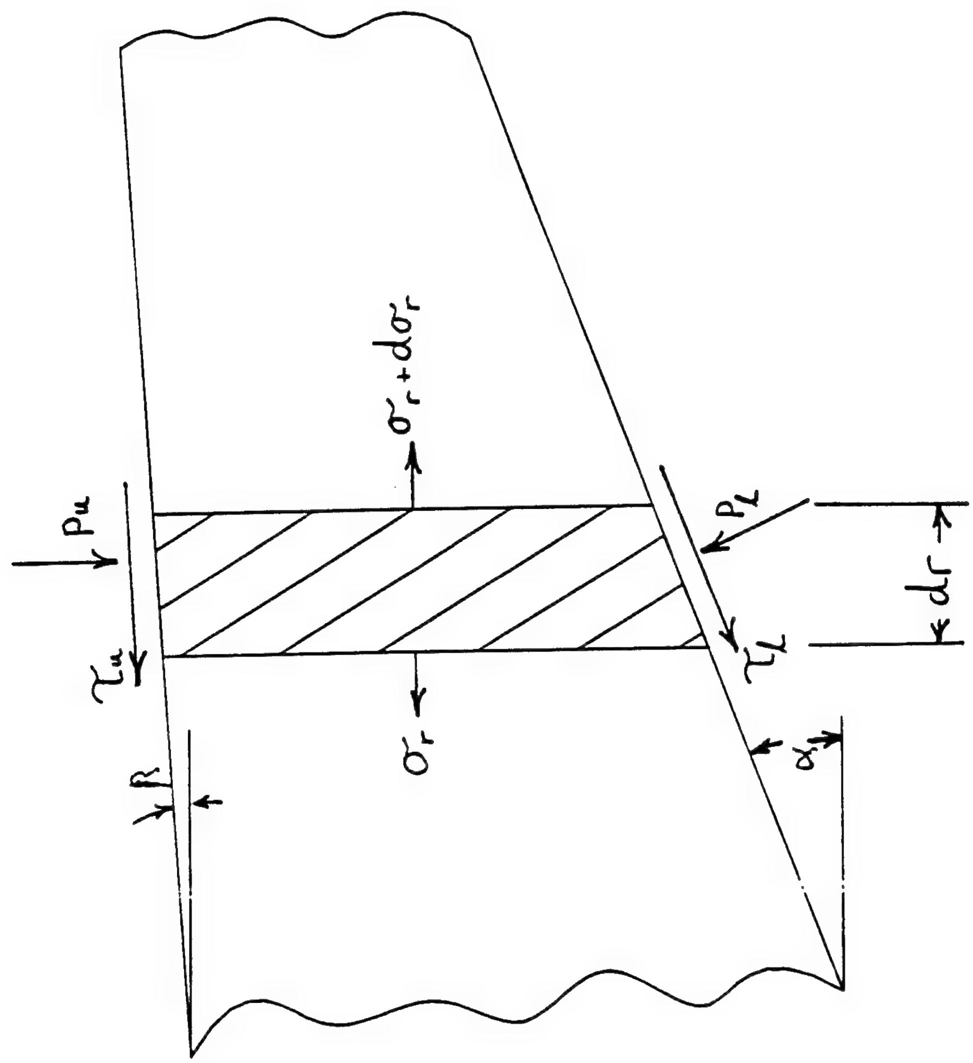
$$\frac{dh}{dr} = (\tan\,\beta - \tan\,\alpha)$$

Coulomb Friction

$$\tau_k = \mu P_u \qquad \tau_l = \mu P_l$$

Sticking Friction

$$\tau_u = mk \qquad \tau_l = mk$$



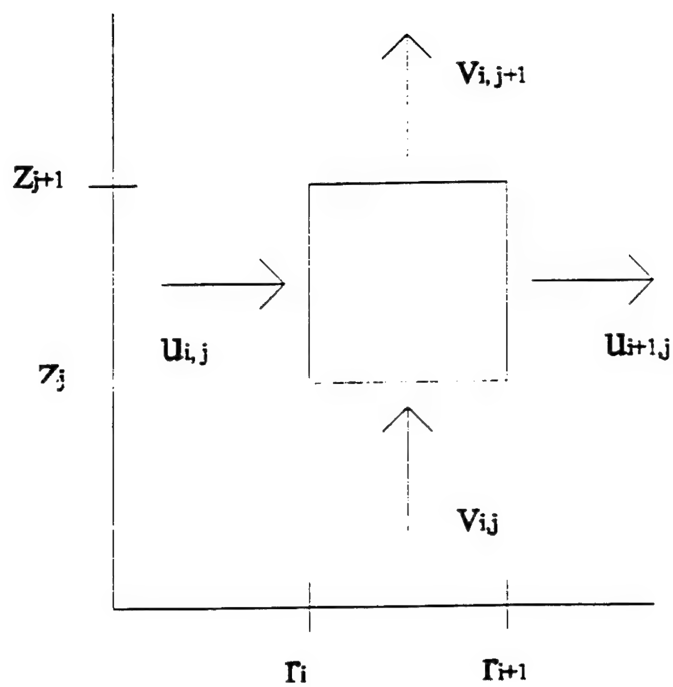
P_u
τ_u
β
σ_r
$\sigma_r + d\sigma_r$
P_l
τ_l
dr
α_1

UPDATED UPPER BOUND ELEMENTAL TECHNIQUE (UBET)

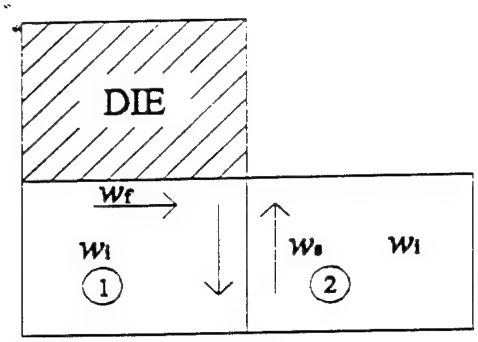
UBET Energy Minimization

UBET Velocity Field Elements

UBET Velocity Fields



UBET Energy Minimization



where

$$\dot{w}_i = \int_V \bar{\sigma}\, \bar{\epsilon}\, dV \qquad \dot{w}_s = k \int_{\Gamma_s} |\Delta U_s|\, ds \qquad \dot{w}_f = mk \int_{\Gamma_s}$$

$\dot{w}_i$ = *Internal Plastic Deformation Power*

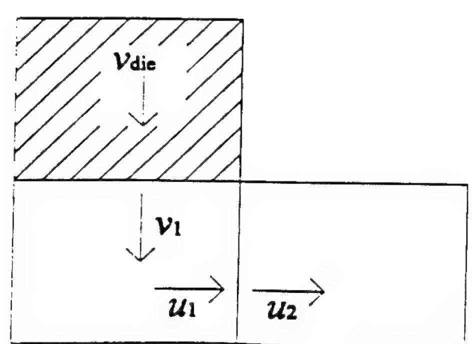
$\dot{w}_s$ = *Shear Losses Along Element Boundaries*

$\dot{w}_f$ = *Friction Losses at Die–Workpiece Interface*

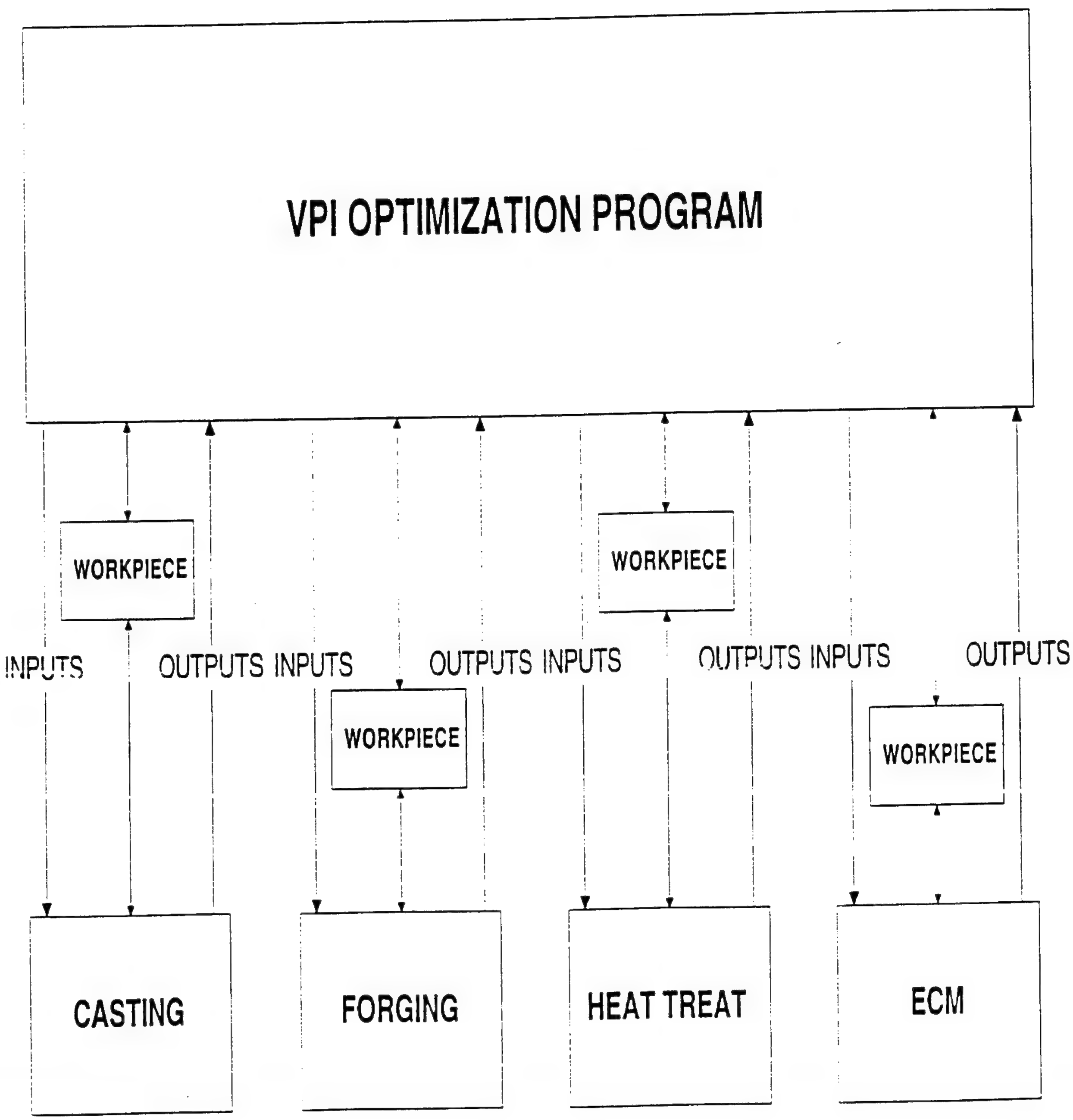
Compatibility

$v_{die} = v_1$

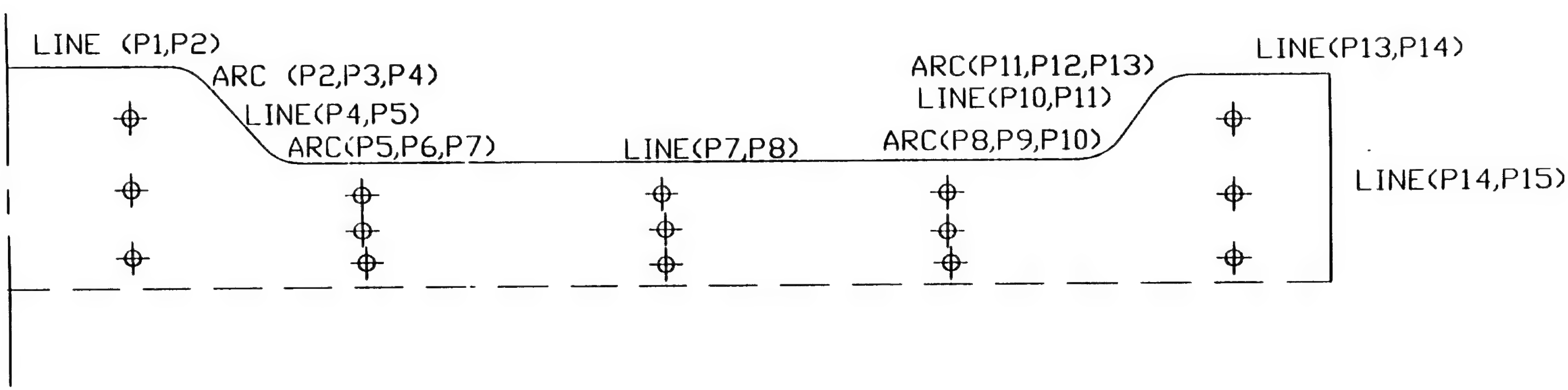
$u_1 = u_2$



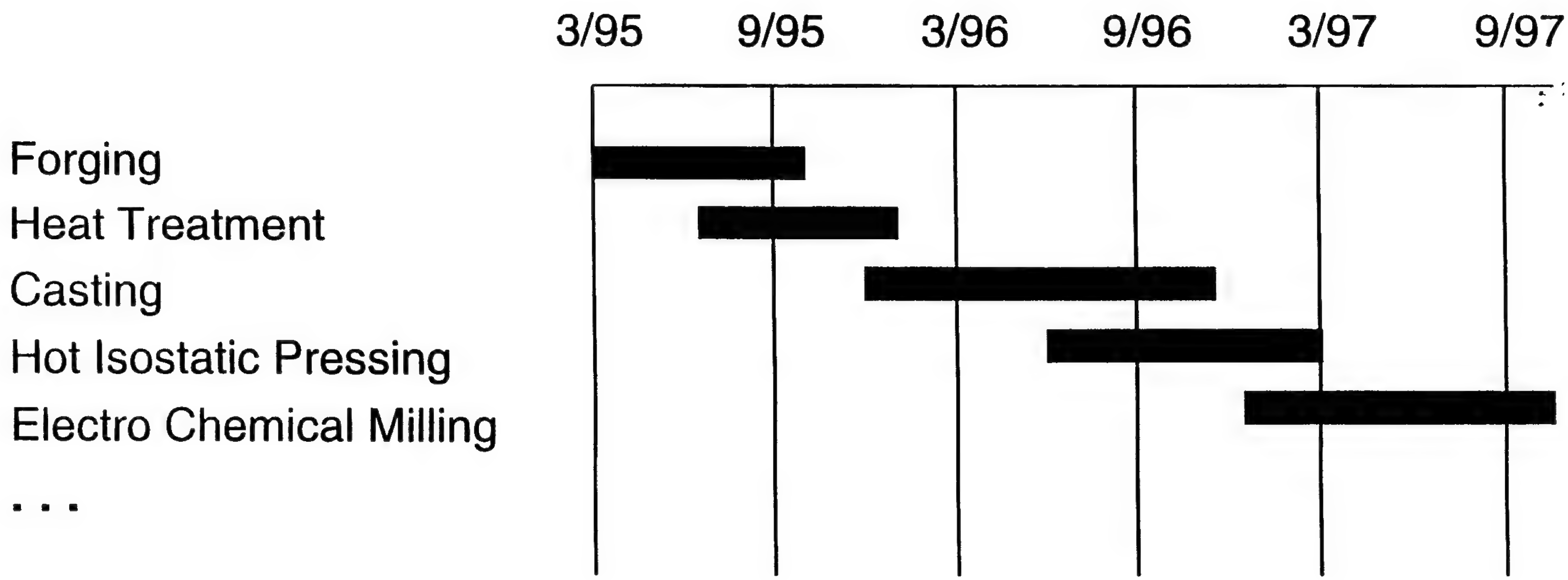
Program Structure



Workpiece Data Structure



Targeted Schedule for Process Model Development



Analyzing Discrete Event Simulation Models of Complex Manufacturing Systems: a Computational Complexity Approach

by Sheldon H. Jacobson and Alan W. Johnson
Virginia Tech

Abstract

Discrete-event computer simulation is a powerful modeling tool for analyzing complex systems that cannot be studied analytically. The value of simulation depends on how effectively and efficiently one can build and utilize the simulation model to gain insights into complex, real-world systems.

The goal of this research is to introduce a new set of tools for building and analyzing simulation models of complex systems, with a particular emphasis on manufacturing processes for turbine engine parts, under development at Wright Laboratory (Materials Process Design Branch). The basic research question to be considered is: Given a complex manufacturing system, where the different processes and operations can be modeled using discrete event simulation model events, is it possible to determine the sequence of processes and operations, hence events, that results in finished products with prespecified attributes and output measures? The attributes and measures of such manufacturing processes typically involve physical and operational properties of the finished products, as well as metrics for affordability, predictability, and repeatability of the outputs.

Research questions of this type are often addressed using adhoc procedures and problem-specific approaches. This project uses the theory of computational complexity to provide a framework to study these research questions, as well as construct new heuristic algorithms to address them. These heuristics involve simulated annealing, genetic algorithms, and tabu search. These three heuristics will be cross-fertilized, resulting in hybrid approaches, as well as enhanced to exploit any special structure offered by the manufacturing problems.

ANALYZING DISCRETE EVENT SIMULATION MODELS OF COMPLEX MANUFACTURING SYSTEMS: A COMPUTATIONAL COMPLEXITY APPROACH†

SHELDON H. JACOBSON, ALAN W. JOHNSON
DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING
VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
BLACKSBURG, VIRGINIA 24061-0118
(703) 231-7099 (OFFICE)
(703) 231-3322 (FAX)
JACOBSON@TIOLI.ISE.VT.EDU
JOHNAL@TIOLI.ISE.VT.EDU

†THIS WORK IS SUPPORTED BY THE NATIONAL SCIENCE FOUNDATION (DMI-9409266) AND THE AIR FORCE OFFICE OF SCIENTIFIC RESEARCH (F49620-95-1-0124)

BACKGROUND AND MOTIVATION

* BUILDING AND ANALYZING DISCRETE EVENT SIMULATION MODELS ARE DIFFICULT TASKS.

* COMPUTER SUPPORT (SOFTWARE) AND COMPUTING POWER (HARDWARE) ARE PLENTIFUL.

* RESEARCHERS HAVE BEEN WORKING ON DEVELOPING A COMPREHENSIVE MODELING FRAMEWORK TO AID PRACTITIONERS BUILD AND ANALYZE SIMULATION MODELS.

DEFINITIONS

* A *SYSTEM* IS A COLLECTION OF INTERACTING ENTITIES DETERMINED BY CERTAIN LAWS AND POLICIES.

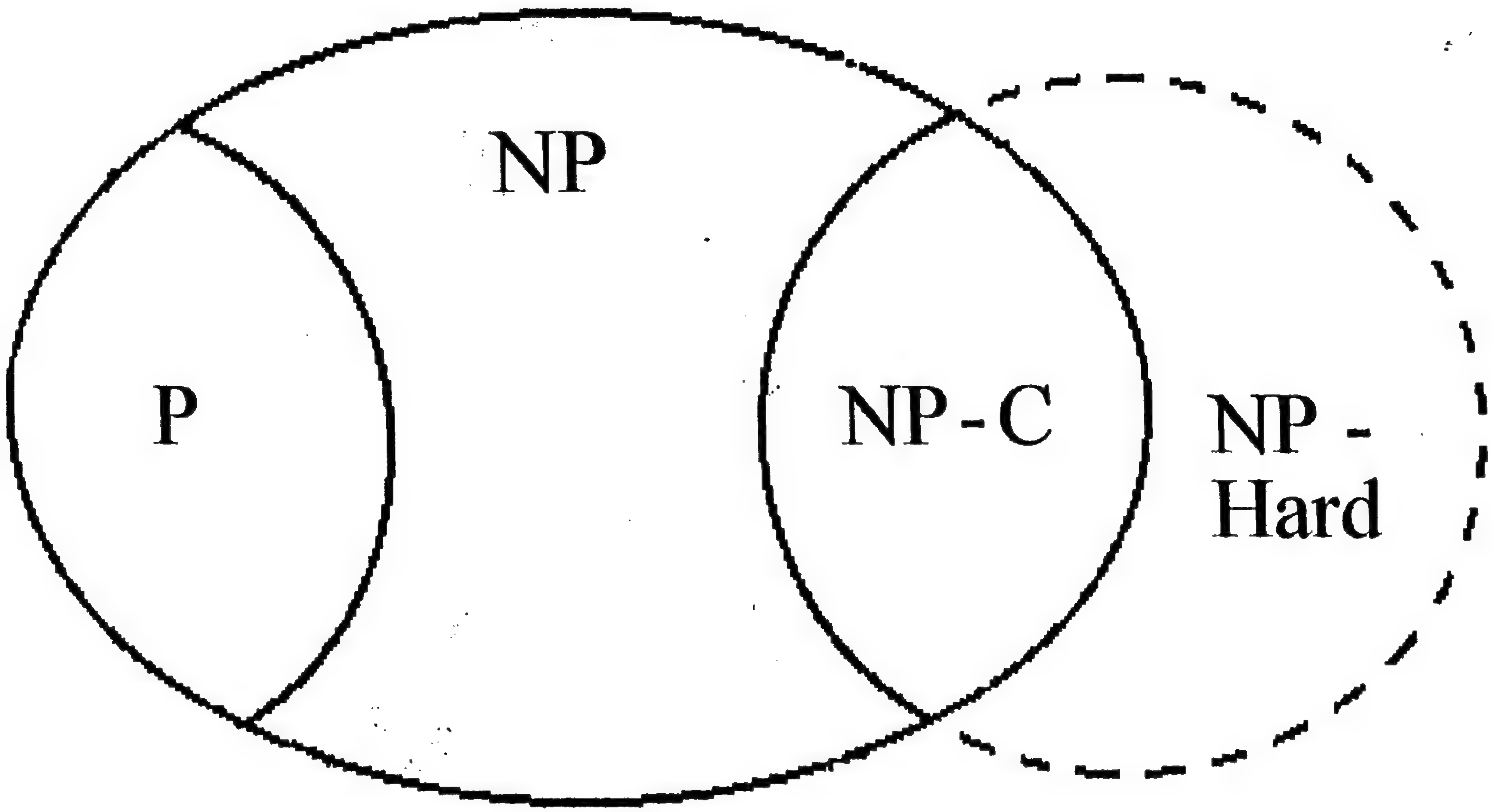
* THE *STATE* IS A COMPLETE DESCRIPTION OF A SYSTEM.

* AN *EVENT* IS WHERE:
  - STATE CHANGES OCCUR
  - OTHER EVENTS ARE SCHEDULED/CANCELLED.

* A *MODEL SPECIFICATION* (MS) IS A SYSTEM REPRESENTATION OF A DISCRETE EVENT SIMULATION MODEL.

* A *MODEL IMPLEMENTATION* (MI) IS A COMPUTER REPRESENTATION OF A MODEL SPECIFICATION.

The World of NP



ACCESSIBILITY:

Instance: A discrete event simulation MS with associated MI
An initial event E_0
A state S
A non-negative finite integer M.

Problem: Find a sequence of events $E_1, E_2, \ldots, E_m$, $m \leq M$ such that the execution of the sequence yields $E_0 E_1 E_2 \ldots E_m \rightarrow S$.

* THIS IS A TRAJECTORY PROBLEM.

* WHY IS THIS PROBLEM INTERESTING?

MANUFACTURING APPLICATION

TURBINE ENGINE MANUFACTURING

* BILLETS ARE TRANSFORMED INTO FINISHED PRODUCTS.

* TITANIUM ALLOYS (TiAl) ARE USED FOR THESE PRODUCTS.

* SEVERAL DISCRETE STEPS (EVENTS) ARE INVOLVED IN THIS TRANSFORMATION:

  - PRECONDITIONING

  - TEMPERATURE VARIATIONS (HEAT TREATMENTS, COOLING)

  - FORGING AND EXTRUSION

  - ???

FUNDAMENTAL RESEARCH QUESTION:

* HOW CAN THESE STEPS (EVENTS) BE SEQUENCED SUCH THAT THE FINISHED PRODUCT MEETS CERTAIN GEOMETRIC, AND MICRO-STRUCTURAL PROPERTIES?

  – AFFORDABILITY
  – PREDICTABILITY
  – REPRODUCIBILITY

SOLUTION:

* FORMULATE THE PROBLEM AS A (DYNAMIC) COMBINATORIAL OPTIMIZATION PROBLEM, WHERE THE DISCRETE OBJECTS ARE SIMULATED EVENTS.

* DEVELOP ALGORITHMS THAT WILL ADDRESS THIS (SEQUENCING) PROBLEM (ACCESSIBILITY).

Steepest Descent Algorithm
(Minimization Problem)

- Select an initial incumbent solution $\boldsymbol{\sigma}$ from solution space $\boldsymbol{\Sigma}$ $(\boldsymbol{\sigma} \in \boldsymbol{\Sigma})$
- Define a cost function $\mathbf{f}(\boldsymbol{\sigma})$ for solution $\boldsymbol{\sigma}$
- Select a solution neighborhood function that defines all neighbors $\boldsymbol{\sigma}'$ of $\boldsymbol{\sigma}$ $(\boldsymbol{\sigma}' \in \eta(\boldsymbol{\sigma}))$
- Select a stopping criterion, **N**, that defines the total number of iterations executed

Set the repetition counter $n = 0$
While $n \neq N$:

> Randomly generate a solution $\sigma' \in \eta(\sigma)$
> Calculate cost function change $\delta = f(\sigma') - f(\sigma)$
> *if* $\delta < 0$, *then* $\sigma \Leftarrow \sigma'$
> *else* $n \Leftarrow n+1$

Source: Eglese, R.W. "Simulated Annealing: A Tool for Operational Research", *European Journal of Operational Research,* Vol 46, pp 271-281, 1990.

Tabu Search Algorithm
(Minimization Problem)

- Select an initial incumbent solution σ from solution space Σ $(\sigma \in \Sigma)$
- Define optimal solution as σ^*
- Define a cost function $\mathbf{f}(\sigma)$ for solution σ
- Select a solution neighborhood function that defines all neighbors σ' of σ $(\sigma' \in \eta(\sigma))$
- Define a list $\mathbf{L}(\sigma)$ of length $\mathbf{k}$ whose elements are called *tabu moves*

1. - Set the repetition counter $n = 0$
   - Initialize $L(\sigma)$ as empty
   - $\sigma^* \Leftarrow \sigma$

2. - *if* $\eta(\sigma) - L(\sigma) = \varnothing$ *then* go to step 4.
   - *else*
     -- $n \Leftarrow n+1$
     -- Select $\sigma'_n \in \eta(\sigma) - L(\sigma)$ such that
       $f(\sigma'_n) = \text{minimum}\{f(\sigma'): \sigma' \in \eta(\sigma) - L(\sigma)\}$

3. - $\sigma \Leftarrow \sigma'_n$
   - *if* $f(\sigma) < f(\sigma^*)$ *then* $\sigma^* \Leftarrow \sigma$

4. - *if* stopping criterion met, or
     if $\eta(\sigma) - L(\sigma) = \varnothing$ (from step 2.), *then* stop
   - *else* update tabu list $L(\sigma)$, and return to step 2.

Source: Glover, F., "Tabu Search - Part 1", *ORSA Journal on Computing*, Vol 1, No. 3, pp 190-206, 1989.

KEY COMPONENTS OF ALGORITHMS

* CONFIGURATION SPACE APPROACH

  - SOLUTION SPACE

    - SET OF ALL POSSIBLE SEQUENCES OF EVENTS (LARGE!)

  - COST (OBJECTIVE) FUNCTIONS

    - MUST CAPTURE:
      - ALL GEOMETRIC AND MICROSTRUCTURAL PROPERTIES OF THE FINISHED PRODUCT
      - KEY CHARACTERISTICS OF THE MANU-FACTURING PROCESS

  - NEIGHBORING SOLUTIONS

    - IDENTIFY MECHANISMS TO EFFICIENTLY TRAVERSE THE SOLUTION SPACE

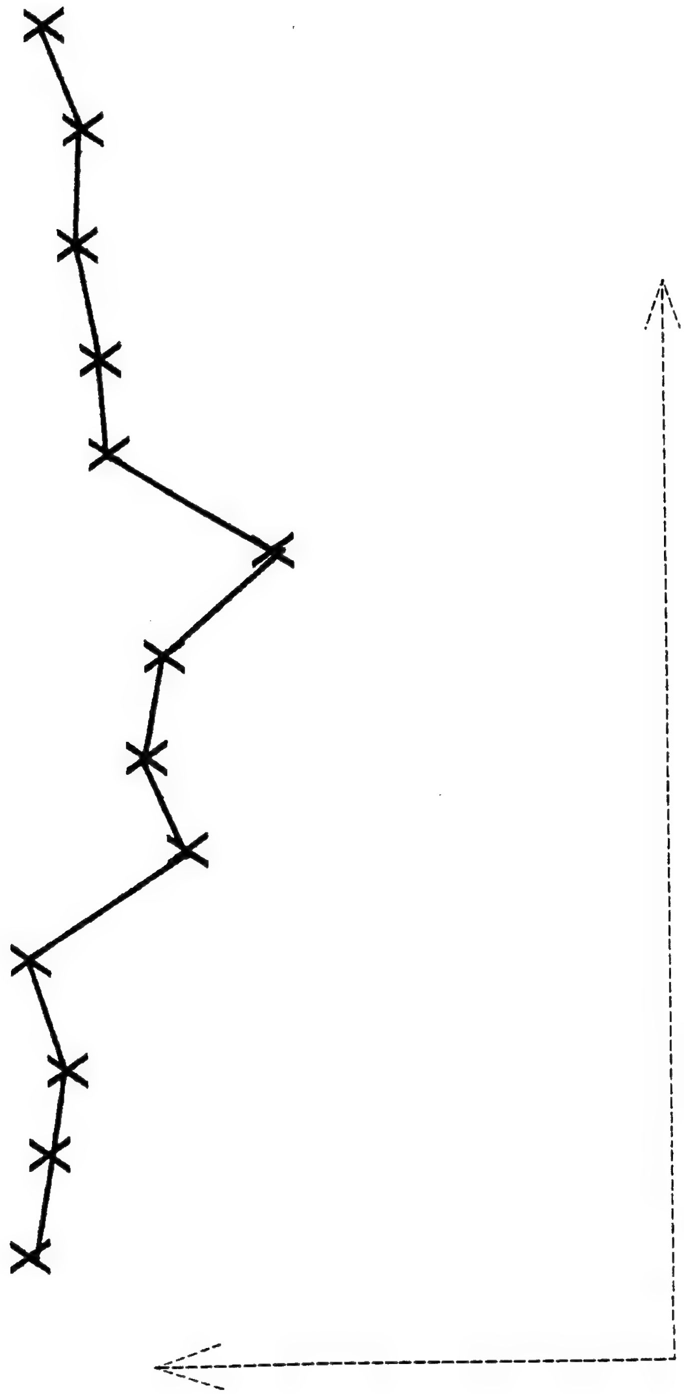
* ALGORITHM RESEARCH ISSUES

- VARIANCE ENHANCEMENT TECHNIQUES & PROBLEM RANDOMIZATION

- TUNNELLING (RANDOM PERTURBATION RELAXATION)

- SENSITIVITY ANALYSIS OF ALGORITHMS

- FALSE NEGATIVE PROBLEM



Simultaneous Design & Analysis

May 23:

12:00-1:30 Lunch

1:30 Finite Element Analysis for Simultaneous (Billet, Die and Part) Design for Deformation Processes
T. Hughes, Stanford

2:15 Automated 3-D Mesh Generation for Deformation Process Design
M. Shepherd, RPI

3:00 Alternative Materials & Processes: Incomplete Design and Multi-process Optimization
A. Chemaly, Wright Laboratory Visiting Scientist

3:45-4:00 Break

4:00 Incomplete Design for Casting & Machining Demonstration
H. AL-Kamhawi, Wright Laboratory Visiting Scientist

Finite Element Analysis for Simultaneous (Billet, Die and Part) Design for Deformation Processes

Thomas J.R. Hughes
Division of Applied Mechanics
Durand Building
Stanford University
Stanford, California 94305

Abstract

This presentation describes a research program directed towards the development of automated design procedures for aluminum extrusion technology. The objective is to eliminate costly trial and error by being able to simultaneously design the product, die, billet, and process (e.g., extrusion temperatures and speeds, uniformizing metal flow, etc.), within constraints of feasibility, and satisfying objectives including, but not limited to, optimizing shape, surface finish, and properties of the product, processing costs, time to market, and full utilization of capabilities. The approach is based on the development of efficient and effective analysis of the whole processing system employing newly developed finite element solution technologies for complex, multiregion, multiphysical behavior. Generalizations of these methodologies to include Arbitrary Lagrangian-Eulerian (ALE) mesh descriptions for nonlinear, elasto-viscoplastic mechanical constitution equations will allow the faithful modeling of the metal flow within the die system and the accurate attainment of final shape upon exit. Automatic meshing and adaptive remeshing will insure efficient and accurate simulation of the entire forming process. New element technologies facilitating the use of general meshing procedures for difficult metal-forming processes involving a variety of kinematical constraints, such as incompressibility, contact, etc., are utilized. Feature based design methodologies, parametric modeling, and knowledge-based engineering techniques will constitute the fundamental methodologies for representing designs, managing the hierarchy of analysis models, performing model reduction and feature removal, and effectively utilizing design knowledge. Modern, three-dimensional, interactive visualization procedures are employed to animate simulations and design evolution. The software will be accessed through a modern and easy-to-use graphical user interface developed for extruders, die makers, and designers. The current status of our software and plans for executing the research and development program will be presented.

Finite Element Analysis for Simultaneous (Billet, Die and Part) Design for Deformation Processes

T. Hughes

Stanford University

75

Outline

- Introduction
- Technical Discussion
    - Singularities
    - Localization of Deformation
    - *hp*-Adaptive Refinement
    - Element Technology
    - Contact
    - Friction
    - ALE

Outline (cont.)

- Technical Discussion (cont.)
  - Space-time Formulation
  - Die Flexibility
  - Region-based Computational Architecture
  - Parallel Processing
  - Equation Solving Technologies
  - Visualization
  - Control of Process

Outline (cont.)

- Leveraging Existing Technology
  - Modeling Environment
  - Mesh Generation and Adaptivity
  - Geometric Modeling
  - Simulation and Visualization
  - User Interface
  - Optimization
- Summary

Introduction

- Research directed towards the development of automated design procedures for aluminum extrusion technology

- Goal: Eliminate costly trial and error by being able to simultaneously design the product, die, billet, and process, within constraints of feasibility and satisfying objectives including but not limited to optimizing shape, surface finish, and other product properties, processing costs, time to market, full utilization of capabilities, etc.

- Approach is based on the efficient and effective analysis of the whole process employing newly developed finite element solution technologies for complex, multiregion, multiphysical behavior

Introduction (cont.)

- ALE mesh descriptions for nonlinear, elasto-viscoplastic mechanical constitutive equations will allow accurate modeling of the metal flow within the die and container and the accurate attainment of final shape upon exit

- New element technologies facilitating the use of general meshing procedures for difficult metal-forming processes involving kinematical constraints, such as, incompressibility and contact, are utilized

- Automatic meshing and adaptive remeshing will be employed to attain efficient and accurate simulations (see presentation of M. Shephard)

Introduction (cont.)

- Feature based design methodologies, parametric modeling, and knowledge based engineering technologies will be utilized to represent designs, manage hierarchies of analysis models, perform model reduction and feature removal, and effectively utilize experience and design knowledge (see presentation of A. Chemaly)

- Modern, three-dimensional, interactive visualization procedures will be employed to animate simulations and design evolution

- The software will be accessed through an easy to use graphical user interface developed for extruders, die makers and designers

Technical Discussion

Singularities

- Geometrical discontinuities within die and at the point of exit give rise to "attached" singularities in the rate-of-deformation field

- These have the structure of classical elliptic singularities

- Approach: p-**adaptivity**

Localization of Deformation

- In flat-faced dies, very strong, localized gradients exist across "shear surfaces," causing a "dead-metal" zone to form in front of the die

- Classical polynomial interpolation is not well-suited to approximate such abrupt discontinuities

- Special ingredients need to be incorporated in a numerical formulation to avoid mesh-dependent pathologies

- Various methodologies have been proposed

- Approach: Combination of h-**adaptivity** and **subgrid-scale modeling** techniques which embody the features necessary to avoid mesh sensitivity

hp-Adaptive Refinement

- Both h and p strategies are relevant in simulating extrusion processes

- h-adaptivity for interior surfaces of near-discontinuity

- p-adaptivity for elliptic singularities and high-accuracy in regions of smooth response

- Approach: **Combined hp-adaptive strategy**

Element Technology

- Incompressible kinematics is fundamental to the description of the billet and part
- Fully automatic meshing and adaptive remeshing are available for tetrahedra
- Need for *hp*-adaptive tetrahedral meshing puts stringent requirements on element technology for extrusion problems
- Most combinations of velocity and pressure interpolations give rise to unstable, hence nonconvergent, numerical approximations within the classical Galerkin formulation
- Approach: **Stabilized methods** which support any combination of velocity and (continuous or discontinuous) pressure (H.-Franca-Stenberg-Brezzi-etc.)

Contact

- Mechanical and thermal contact between billet and container and die plays a significant role in the extrusion process

- The mathematical sensitivities are similar to that for incompressible behavior

- The standard treatment of contact within the Galerkin method does not provide a suitable basis for *hp*-strategies

- Approach: **Stabilized methods** (H.-Barbosa)

Friction

- Friction is an important aspect of billet-container-die interaction

- Friction is an important source of heat generation

- Coulomb friction is viewed as inadequate for representing extrusion processes

- Approach: Fundamental study in comparing mechanical and thermal implications of various friction models on the extrusion process

ALE

- Solid mechanical approaches have been unable to deal with the enormous deformations (i.e., "flows") associated with extrusion problems

- The reason is the reliance upon the **Lagrangian** description

- Mesh entanglement requires frequent remeshing throughout the extrusion process

- ALE (arbitrary Lagrangian-Eulerian formulation) has been used primarily for fluid mechanical phenomena to simulate **free-surface flows** and **fluid-structure interaction**

- One of the **fundamental** and **absolutely essential** technologies for extrusion simulation

ALE (cont.)

- Constitutive integration

  - Extremely simple for typical Newtonian and certain classes of non-Newtonian constitutive equations

  - More complicated for history-dependent constitutive models, nevertheless widely employed in chemical engineering for **polymer extrusion**

  - Stabilized methods widely used in polymer extrusion

Space-time Formulations

- New, potentially superior ALE technology has been developed in recent years from a **space-time** viewpoint

- Conservation laws automatically enforced, a difficulty with some traditional ALE methods

- Comprehensive mathematical theory exists

- A rigorous basis for *a posteriori* error estimation and hence adaptive refinement strategies

- Efficiency trade-offs with traditional ALE

Die Flexibility

- It is essential to model the flexibility and heat conduction behavior of the container and die

- Requires a multiregion thermomechanical contact algorithm incorporating a flow-like ALE description of billet and part in conjunction with a classical Lagrangian description of the container and die

- Also requires a multiphysics solution strategy

Region-based Computational Architecture

- The system is composed of distinct physical regions

- Requires a computational architecture supporting distinct physical and mathematical models, and their interaction through interfaces

- Separate regions also support and simplify adaptivity and instantiation of design changes

- Rapid progress in hardware technology and practical necessity suggest extrusion simulations will be done at the desktop

- For the present, these calculations will be CPU-intensive and will benefit from the use of powerful parallel systems

- Two-level parallel architecture

  - Top-level adopts the **message passing** paradigm on a domain decomposition of the physical problem

  - Second level employs the **data parallel** paradigm within the subdomains

  - Either level of parallelism can be absent

  - Enables efficient performance to be attained on a broad array of parallel architectures

  - Operational on existing platforms, such as IBM SP-2, CONVEX Exemplar, SGI Power Challenge, Thinking Machines CM-5, etc.

Equation Solving Technologies

- A variety of parallelized iterative and direct equation solving technologies will be employed

- Continual progress is being made in this area

Visualization

- Modern, three-dimensional interactive computer graphics will play an essential role in the visualization of process and design evolution

- Animation and archiving capabilities will enable a physical understanding to be attained of very complex phenomena

Control of Process

- Code architecture needs to be able to accommodate user-friendly programmable process control

Leveraging Existing Technology

Modeling Environment

AML (Advanced Modeling Language) from Technosoft, Inc.

- Feature-based design methodologies
- Parametric modeling
- Knowledge-based engineering techniques
- Represent designs and processes
- Manage hierarchies of analysis models
- Perform model reduction and feature removal
- Effectively utilize design information

Finite Octree from RPI

- Interfaced to AML
- Automatic, unstructured mesh generation
- Adaptive remeshing

Shapes from XOX, Inc.

- Interfaced to AML
- Provides a powerful non-manifold topology environment for building analysis models

Spectrum Multiphysics Solver and Visualizer from Centric Engineering Systems, Inc.

- Extensive multiphysics, multiregion parallel simulation and interactive visualization capabilities
- Integrated with AML, Shapes and Finite Octree
- Arbitrary Lagrangian-Eulerian meshing capability for billet and part
- Three-dimensional free surface capability
- Extensible, high-level platform for research and development

User Interface Toolkit and Optimization Kernel

ILOG Views from ILOG

- Object-oriented, graphical user interface toolkit
- Customized interfaces for end users, e.g., extruders, die makers and designers

Optimization Kernel

- To be selected when the research reaches a sufficiently mature stage

Summary

- Presently the design of the die for extrusion and the selection of process parameters is done with little or no automation

- This process relies heavily on previous experience and often involves lengthy and costly trial and error cycles

- A **Simulation Based Design** system for extrusion needs to address the complete thermomechanical interaction of the billet, die and part

- Successful development of this methodology will minimize the process design cycle and lead to major time and cost savings in product development

Automated 3-D Mesh Generation for Deformation Process Design

by Prof. Mark S. Shephard
Scientific Computation Research Center
Rensselaer Polytechnic Institute, Troy, NY 12180--3590

Abstract

The use of advanced simulation tools by metal forming process designers is hampered by the complexity of developing and controlling the numerical analysis models required. The availability of automatic mesh generation capability linked directly with the design geometry and reliable analysis methodologies is critical to the elimination of this bottleneck. This presentation focuses on techniques being developed to automatically generate and control three-dimensional finite element meshes for use in deformation process design driven by CAD geometric models.

An automatic mesh generator can be defined as an algorithmic procedure which can create, under program control and without user input or intervention, a valid mesh, or geometric triangulation, for geometric models of arbitrary complexity. The development of algorithmic procedures for the automatic generation of finite element meshes of general three-dimensional geometric models has been an active area of research for several years with procedures which can work with CAD representations now becoming available. The key to the development of an automatic mesh generator is an understanding of the requirements of geometric triangulation and to ensure that they are addressed by the mesh generator.

The 'Finite Octree' procedure has been developed to fully meet the requirements of an automatic mesh generator. Finite Octree generates a mesh in a two-step discretization process. In the first step the geometric domain is subdivided into a set of discrete cells which are stored in a regular tree structure. In the second step the individual cells within the Finite Octree are discretized into finite elements, with specific care to ensure the proper matching to the elements in the neighboring cells. All of the individual operations performed ensure the result is a valid discretization of the original domain at that point in the process.

An important aspect of an automatic mesh generator is the ability to exercise control over the mesh gradation in both an a priori and a posteriori sense. Mesh control functions which are spatially-based

provide the most flexibility. The structure of the Finite Octree mesh generator provides spatially-based mesh control since mesh gradation is controlled by the tree levels - with each octant in the tree corresponding to a known portion of the domain geometry. Automatic mesh generators must interact with the geometric representation of the domain being meshed. One approach to provide such information is an interface that directly employs the functionality of the geometric modeling system in the mesh generation process. Finite Octree employs a functional interface built on the geometric modeling system. This approach is consistent with the functional interface methods of the CAM-I Application Interface specification.

Finite Octree's functional interface uses 21 "point-wise" geometric interrogations. The integration of Finite Octree with a new geometric modeling system requires the creation of procedures which request the geometric modeler to perform the 21 interrogations. The approach has been used to link Finite Octree to geometry systems such as Parasolids (which is embedded in the Unigraphics CAD system), Shapes (a product of XOX which is embedded in many design and analysis CAD products) and ACIS (a product of Spatial Technologies which is also embedded in many design and analysis CAD products).

The availability of an automatic mesh generator does not ensure acceptable accuracy of a finite element analysis. However, the careful combination of an automatic mesh generator coupled with an adaptive analysis system can accept as input a geometric representation of the domain, with loads, material properties and boundary conditions applied to the geometric model, and automatically produce solution results to a prespecified degree of accuracy.

During the adaptive analysis of an evolving geometry problem, factors that may indicate a desire to modify the mesh include the need to improve the distribution of elements for effective control of the discretization error or to account for degradation in element shapes due to large deformations. Tools which can address one or both of these needs include (i) node point repositioning, (ii) local refinement and de-refinement procedures, and (iii) remeshing. All three have a role in an efficient metal forming simulation procedure.

Node point repositioning procedures change the mesh by repositioning the grid points while maintaining the given mesh topology. Any effective

Adaptive Lagrangian Eulerian (ALE) procedure must have a node point repositioning procedure for the positioning of the grid after each step. Procedures for positioning of the grid have also been used with Lagrangian techniques as a simple remeshing procedure. Criteria used to reposition nodes can consider minimizing the discretization error (re-refinement), controlling element shapes, or a combination of both.

A number of approaches are available to locally refine and de-refine meshes (i.e., hexagonal 3-D mesh) to control the discretization. When simplex meshes (i.e., tetrahedral 3-D mesh) are used there are a number of algorithmic approaches that avoid the introduction of hanging nodes (which require the introduction of constraint equations). An edge-based procedure, with local swapping provides the highest degree of flexibility in this process. Remeshing procedures have been used in two-dimensional metal forming simulations to both control element shapes and to reflect refinements for discretization control. Automatic remeshing has the same basic requirements as the automatic generation of the initial mesh. The added functionalities required are ensuring the automatic mesh generator can reflect the spatially-based mesh gradations defined by the error estimators, and the ability to update the geometric representation (as needed for meshing) based on the analysis results of the last solution step before a remesh.

AUTOMATED 3–D MESH GENERATION FOR DEFORMATION PROCESS DESIGN

M.S. Shephard

Scientific Computation Research Center
Rensselaer Polytechnic Institute
Troy, NY 12180–3590

Outline

- ☐ Automatic mesh generation with Finite Octree
- ☐ Introduction to automated adaptive procedures
- ☐ Model control issues associated with forming simulations
- ☐ Model building and meshing example for extrusions

INTRODUCTION

Application of advanced simulation tools by designers is hampered by complexity of developing and controlling the required meshes

Over the past several years tools that address this issue have been developed

In particular:

- Automatic mesh generators linked directly to CAD geometry
- Basic adaptive mesh update procedures

In 3–D process design some issues remain

- Appropriate matching of finite element analysis and modeling technologies
- Controlling evolving geometries

AUTOMATIC MESH GENERATION

Goal: To generate a valid discretization in a controlled manner for use in a numerical analysis procedure

Automatic Mesh Generator: A procedure that can generate a valid discretization for geometric domains of arbitrary complexity

An automatic mesh generator

- must be able to operate with no input past the interaction with the geometric model
- should provide devices to control the distribution of elements

Geometric Domain, $G = \{\mathcal{G}, T\}$

Primary Topological Entities, $T = \{{}_{G}T^{0}, {}_{G}T^{1}, {}_{G}T^{2}, {}_{G}T^{3}\}$

DETERMINING MESH VALIDITY

Determining validity of mesh generated by automatic mesh generator critical to its operation

Two basic approaches:

- Carry out each step ensuring its validity
- Generate triangulation and apply assurance algorithm

Second approach required if validity not maintained for each step

Many triangulation algorithms require application of second method

Essentially same effort needed for either approach

GEOMETRIC TRIANGULATION

Definition: Given a set P of M unique points, each classified with respect to the geometry G, an n-dimensional geometric triangulation, $\mathcal{T}_G^n$ is a set of N nondegenerate elements $s_i^{d_i}$

$$\mathcal{T}_G^n = \left\{ s_1^{d_1}, s_2^{d_2}, s_3^{d_3}, \ldots, s_N^{d_N} \right\}$$

with $0 \leq d_i \leq n$, satisfying the following properties:

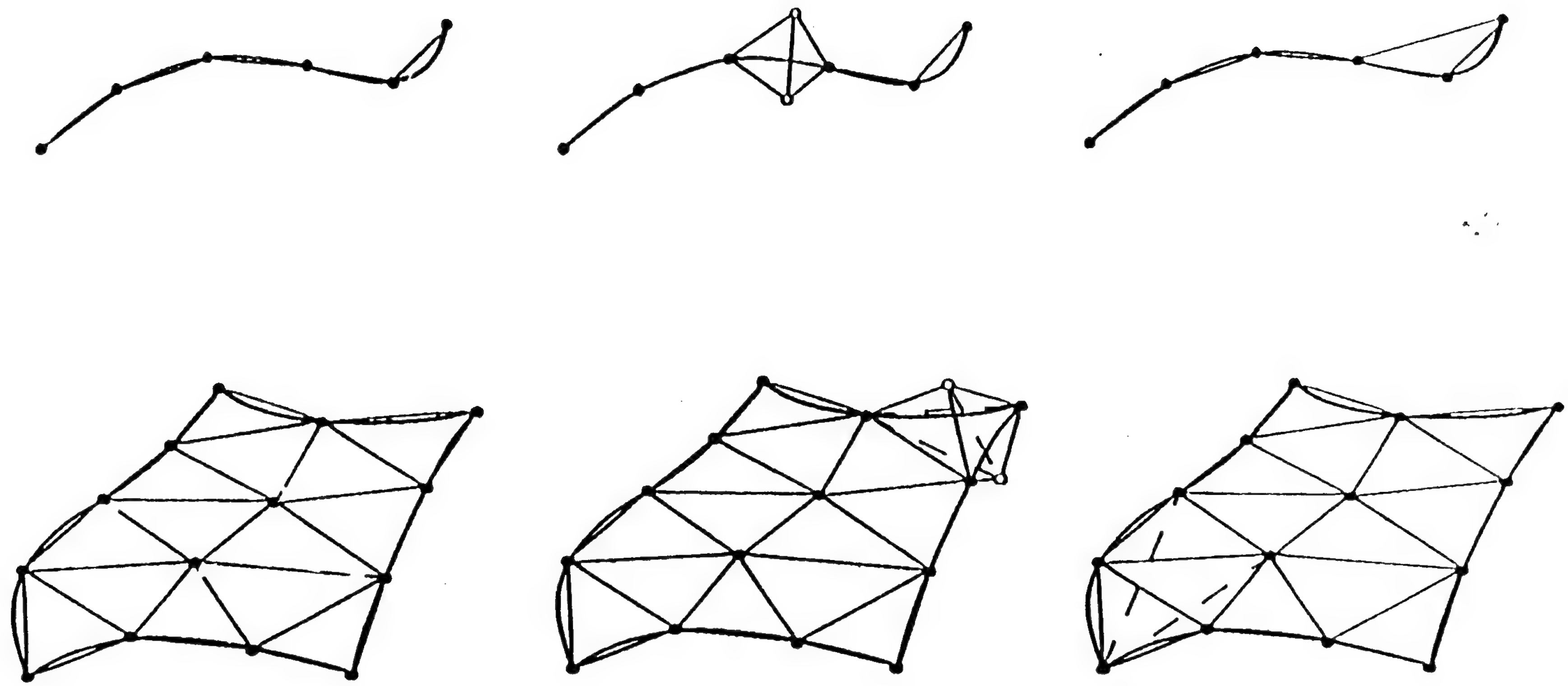
i. All vertices of each $s_i^{d_i} \in P$

ii. For each $i \neq j$, $interior\,(s_i^n) \cap interior\,(s_j^n) = 0$

iii. $\mathcal{T}_G^n$ is topologically compatible with G

iv. $\mathcal{T}_G^n$ is geometrically similar to G

TOPOLOGICAL COMPATIBILITY



compatible

topological hole

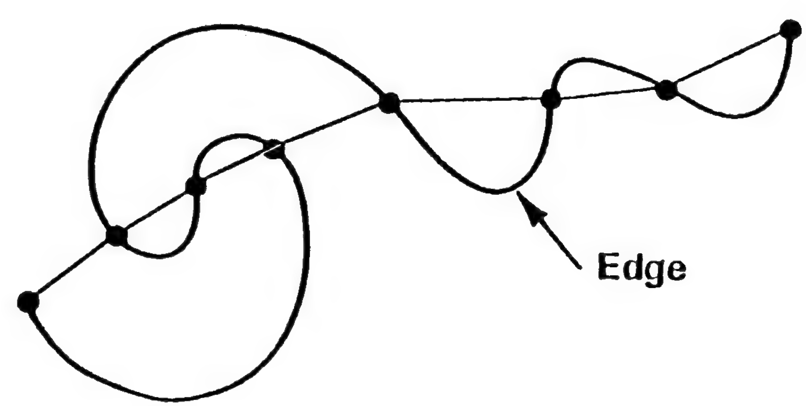
topological redundancy

GEOMETRIC SIMILARITY

Definition: A set of mesh entities $_{M}T^{k}$ is geometrically similar to a model entity $_{G}T_{i}^{k}$ when $_{M}T^{k}$ consist of the N mesh entities

$$_{M}T^{k} = \bigcup_{j=1}^{N} {_{M}T_{j}^{k}}$$

which are classified on the model entity $_{G}T_{i}^{k}$ and the "parametric intersections" of any two mesh entities in the set is null.



COMPATIBILITY ASSURANCE

Convert a triangulation to a Geometric Triangulation

Starting point is a mesh of at least the convex hull

Global compatibility and geometric similarity gained through local classification and compatibility checks coupled with local corrective procedures

Overall process must be bottom-up from lower order model entities to higher order entities

Computational efficiency gained using top-down procedure on each model entity working from its boundary inward

Can use general algorithm for all entities, however, a tailored approach introduces computational efficiency

OCTREE-BASED MESH GENERATION

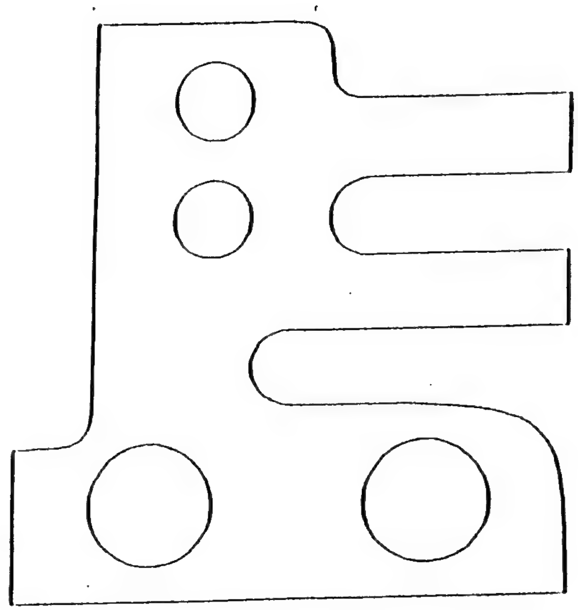
Features common to octree mesh generators:

- Octree structure used to localize processes
- Two level discretization process; 1) geometric model decomposed into octree, 2) octree decomposed into finite element mesh
- Cells (octants) containing portions of object's boundary receive specific consideration to deal with the actual object boundary
- Cell corners and the intersections points defined by the intersection of object and cell boundary entities used as finite element nodes
- Mesh gradations controlled by varying the level of the octree, and thus the cell size, through the object's domain

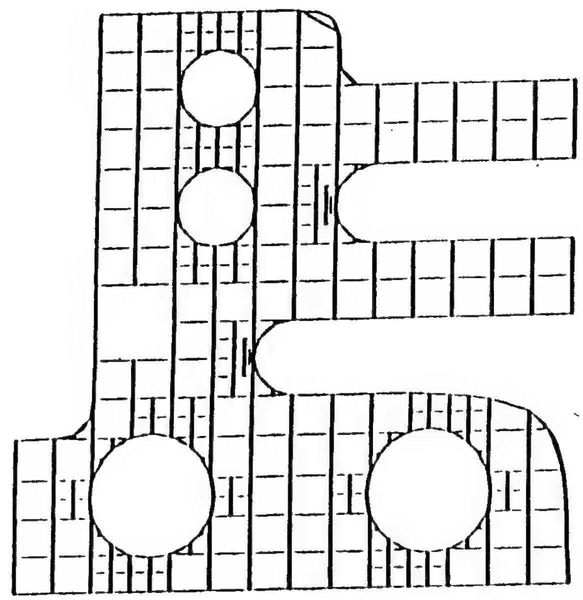
INTRODUCTION TO FINITE OCTREE

- ☐ Creates without user's intervention a valid mesh for geometric model of arbitrary complexity
- ☐ 2 step discretization process:
  - Geometric domain is discretized into a set of cells (terminal octants) which are leaves of an octree structure
  - (Truncated) terminal octants are discretized into Finite Elements
- ☐ Mesh gradation is controlled by varying the octant levels
- ☐ (Inside) terminal octant corners and vertices resulting from the interaction of octant edges (faces) with model faces (edges) are used as mesh vertices
- ☐ Interaction with geometric modeling engine is limited to:
  - Topological adjacency queries
  - Point wise geometric interrogations

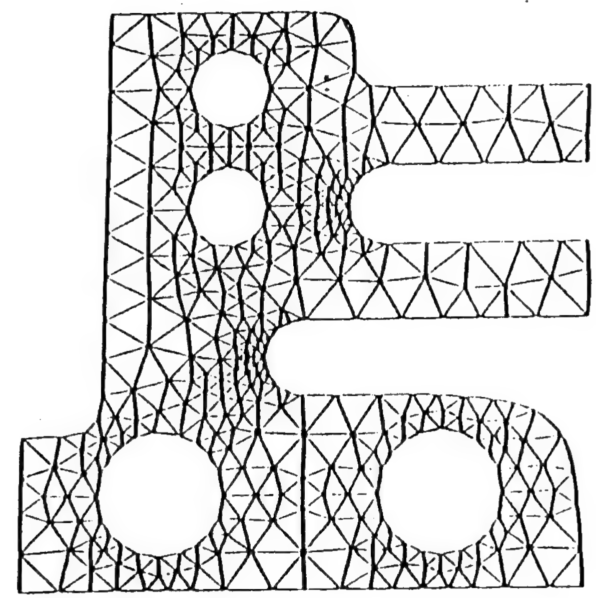
Finite Quadtree



Model

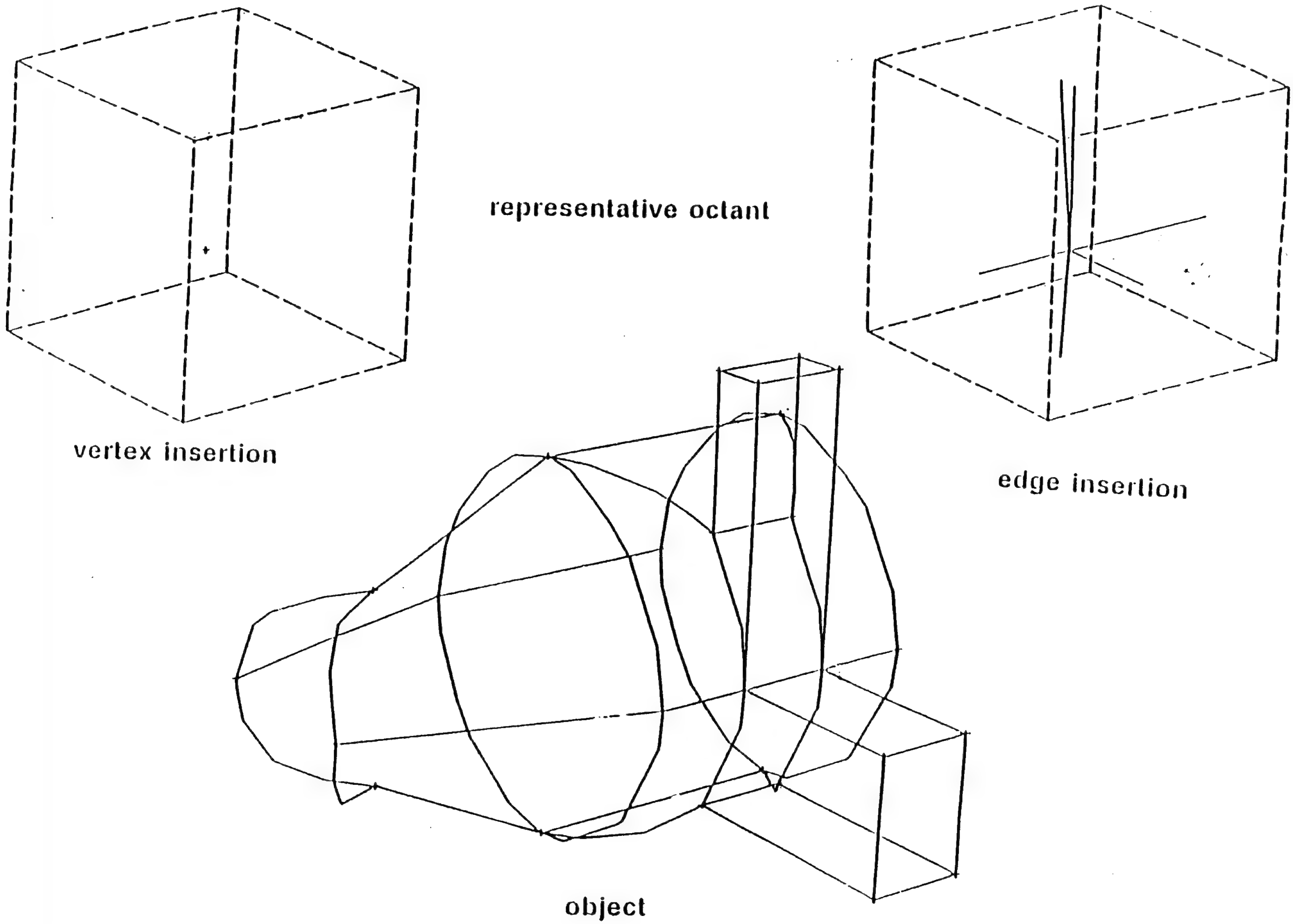


Tree

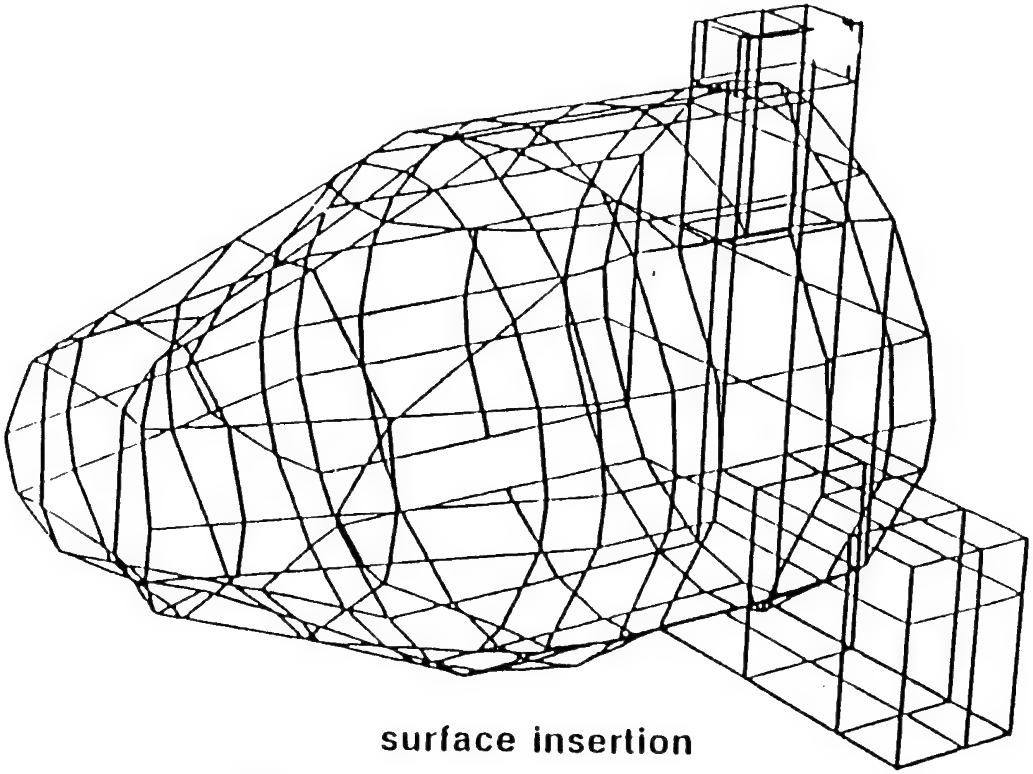


Mesh

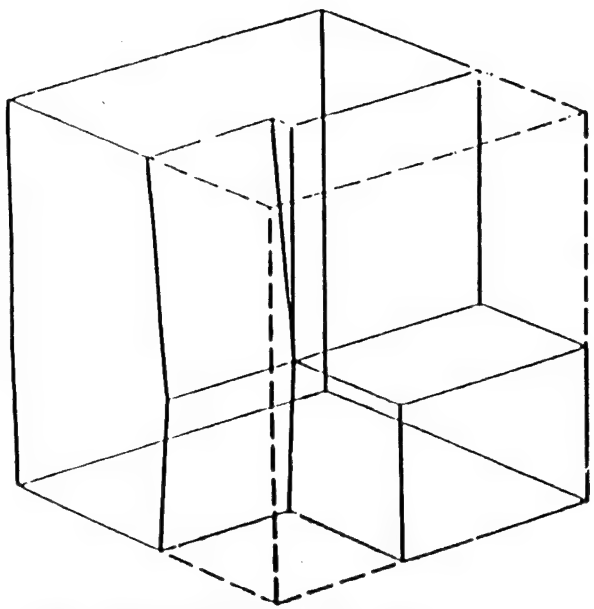
DEFINITION OF FINITE OCTREE



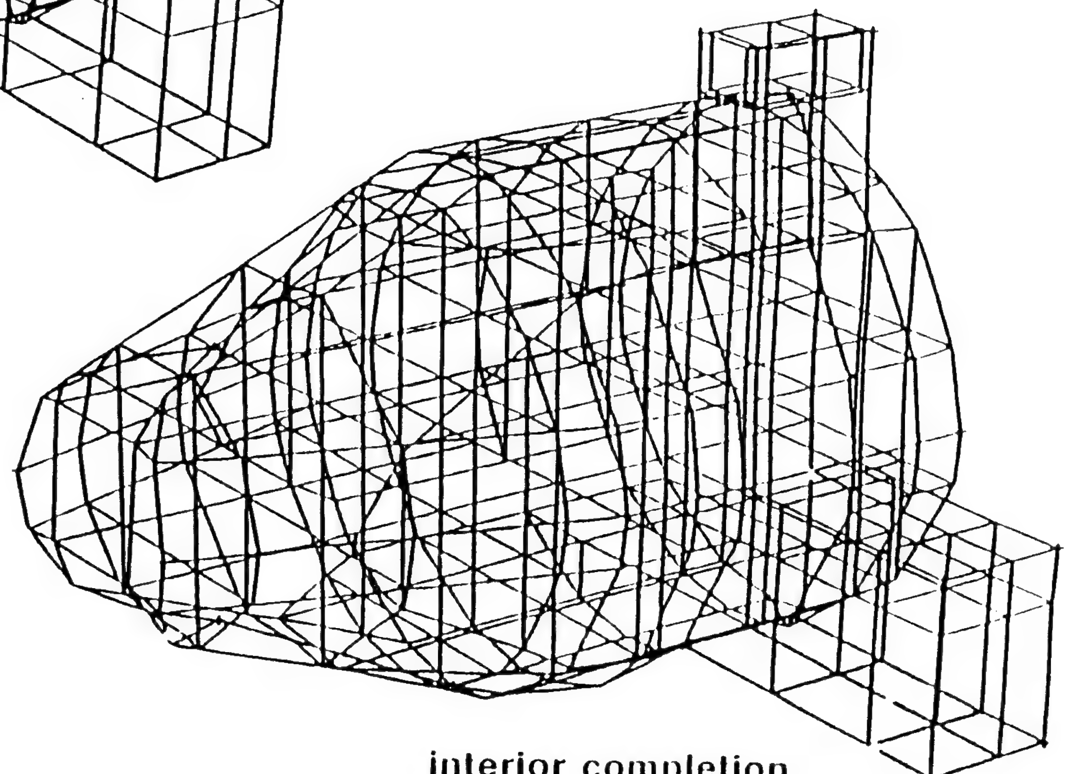
DEFINITION OF FINITE OCTRI



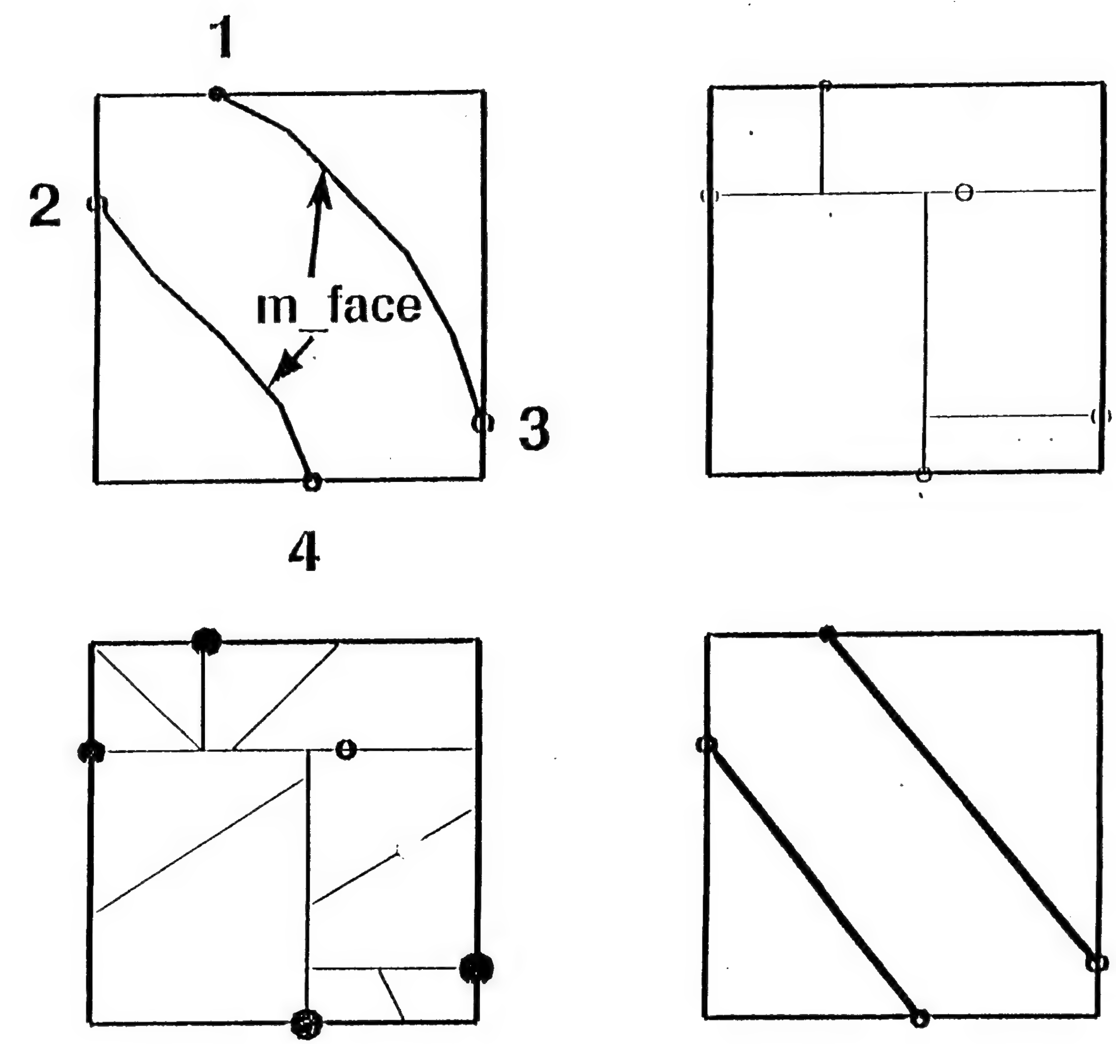
surface insertion



representative octant



interior completion

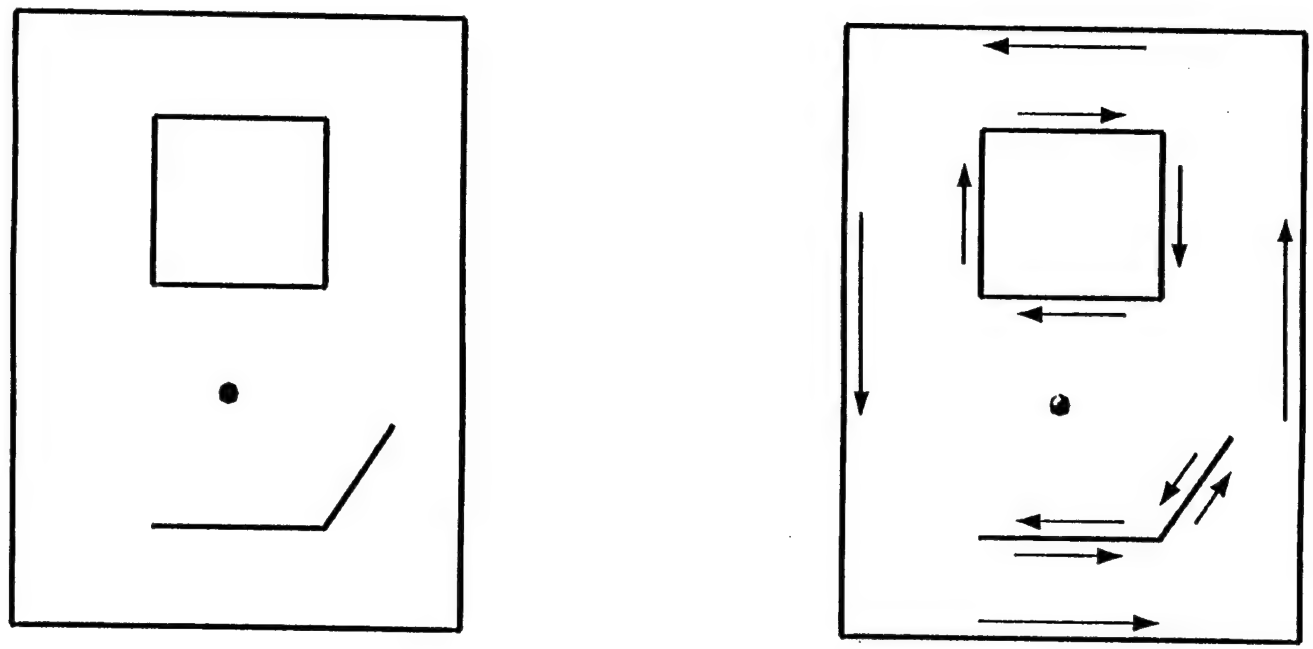


Binary tree on an octant face

LOOP BUILDING ALGORITHM

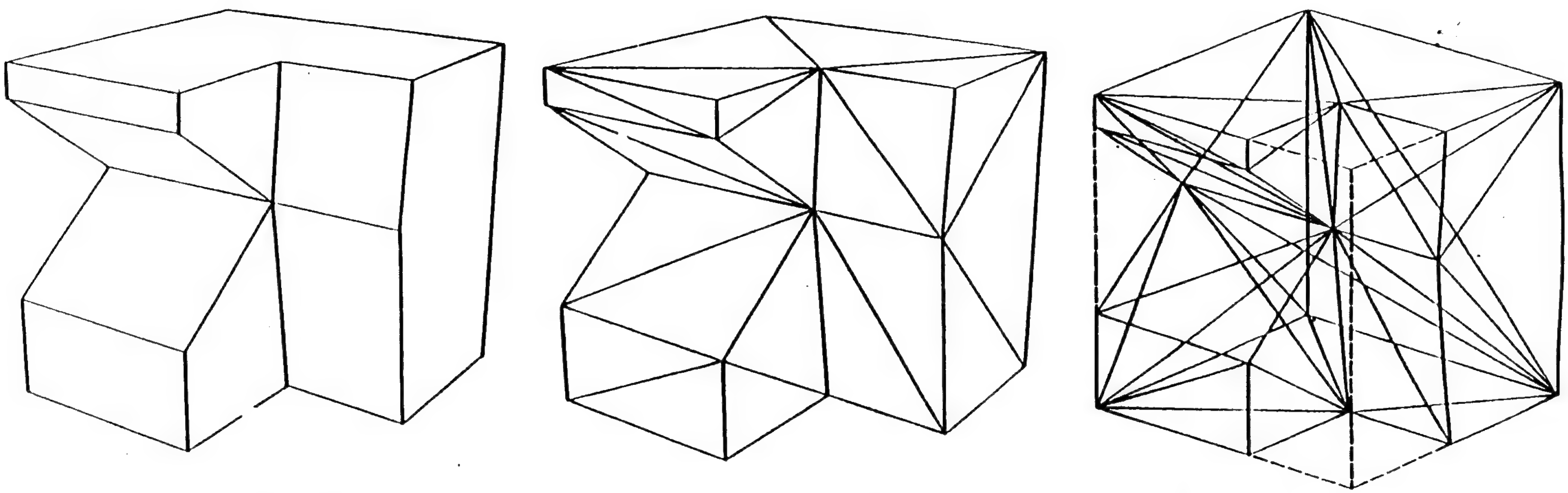
- Result of interaction of a model face with octant boundaries is a list of mesh edges classified on the closure of the model face
- Must construct mesh loops to provide discrete representation of intersection with the model face
- Building 3–D loops for non-manifold models is complex
- Key constructs used to support the construction of loops are:
  - Classification of mesh edges with respect to model and octants
  - Orientation of model edges with respect to model face
  - Classification and type of loops with respect to model face
  - Mesh and model adjacencies
  - Parametric spaces of model faces (when available)

- Create a vertex loop for each ${}_M T_i^0 \in {}_M^0T^*$ and ${}_M T_i^0 \sqsubset {}_G T_l^0$ and ${}_G T_l^0 \equiv {}_G T_n^{2'}$ and ${}_M^0 T_i^{**} = \emptyset$ is added as ${}_M T_n^{2'} \in \left\{ {}_M T^{2'} \right\}$



Three loops of edge-uses and one vertex loop are created

OCTANT TRIANGULATION



a) octant geometry b) surface triangulation c) solid triangulation

104

ELIMINATION OF SMALL SEGMENTS

Attempts to collapse finite element entities after the meshing process is complete

Mesh vertices in the vicinity of octant boundary are collected

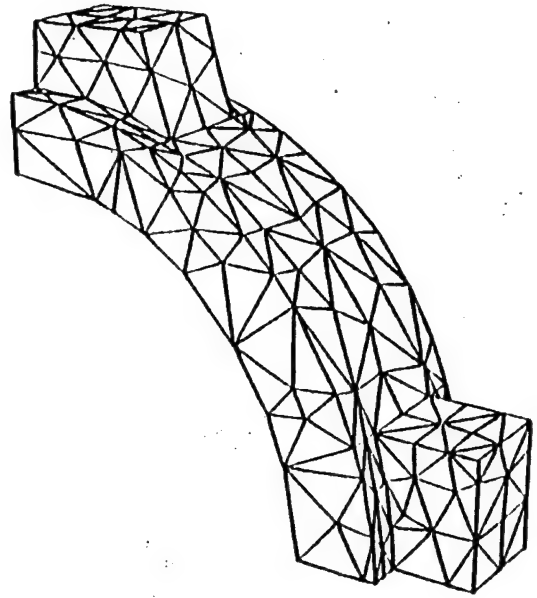
Mesh entities within a distance of 0.2 octant size are eliminated

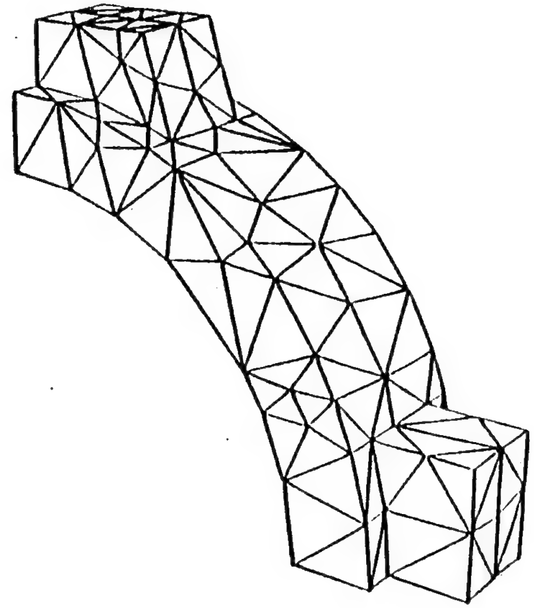
Three types of collapsing are performed:

1. Vertex-Vertex collapsing

2. Vertex-Edge collapsing

3. Vertex-Face collapsing





	Without DRep	with DRep
# Nodes	235	115
# Elements	642	261
Smallest dihedral angle	4.94	21.91
Largest dihedral angle	164.48	140.96
Worst shape	0.119×10^{-3}	0.109
Aspect Ratio	14.61	3.94

Statistics for connecting rod model

NODE POINT REPOSITIONING

Used to improve the shapes of elements

Commonly used centroid (Laplacian) iteration can make element shapes worse and even invalid on unstructured meshes

Constrained centroid iteration much slower and still not guaranteed to improve element shape

Explicit shape improvement iteration can be used

Poorly shaped elements typically near the boundary

Two step procedure developed focusing on the boundary and poorly shaped elements

INTERACTIONS BETWEEN GEOMETRIC MODEL AND MESH GENERATOR

Mesh generators interact with computerized representation of domain being meshed

Geometric modeling systems are used to define these computerized descriptions

Automatic mesh generator must probe the geometric representation to operate

Mesh generator should be independent of geometric representation and should be able to operate on non-manifold geometric domains

FINITE OCTREE INTERFACE TO GEOMETRIC MODELERS

- Interface consists of 22 geometric operators
  - 10 operators returning topological adjacency
  - 2 operators returning meshing attributes
  - 10 operators returning pointwise spatial information
- Topological representation is preprocessed and loaded into Radial Edge data structure, topological operators use it
- Operators returning spatial information use modelers Kernel
- Wrappers are used to allow calling C from FORTRAN on different platforms
- Time required to write the interface to first modeler was 4 months, subsequent interfaces took 2 months

Elimination of Small Mesh Entities

Goal

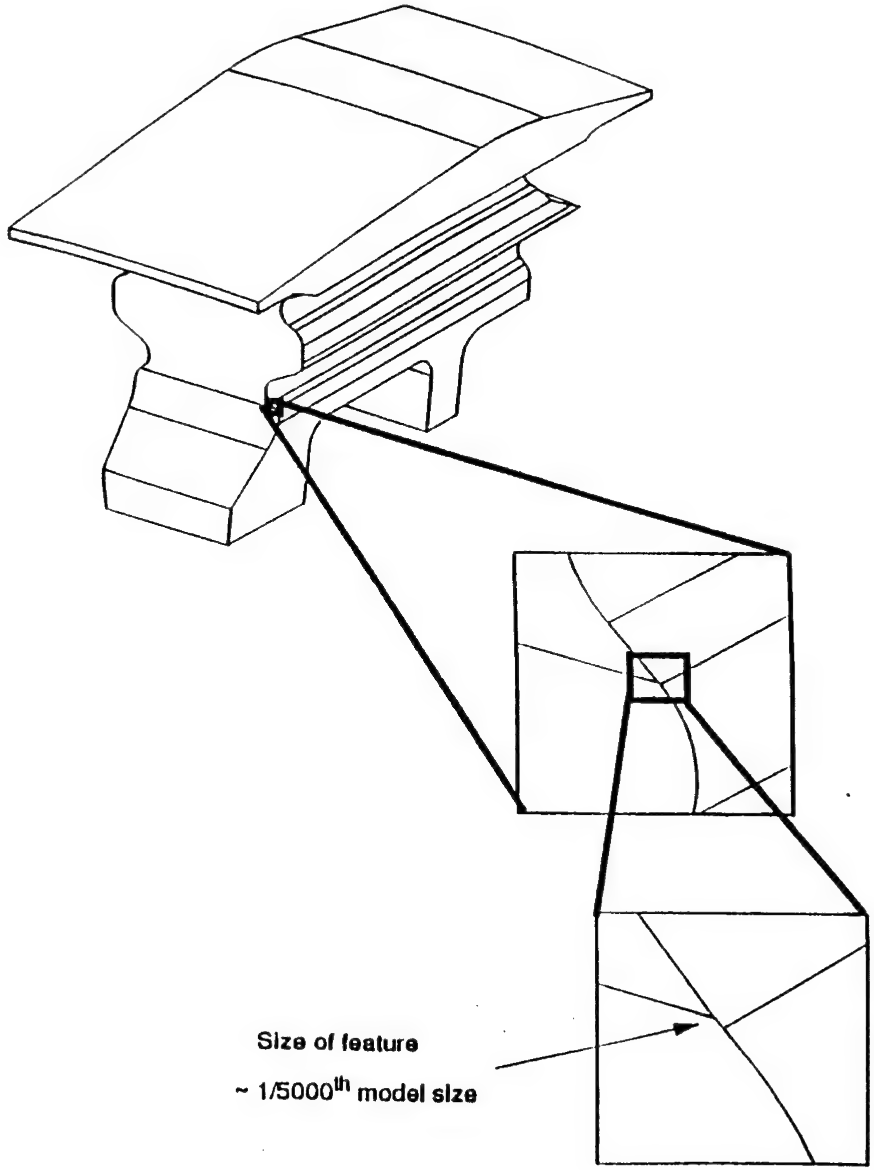
Eliminate “small” mesh edges and “flat” mesh faces resulting from small model features.

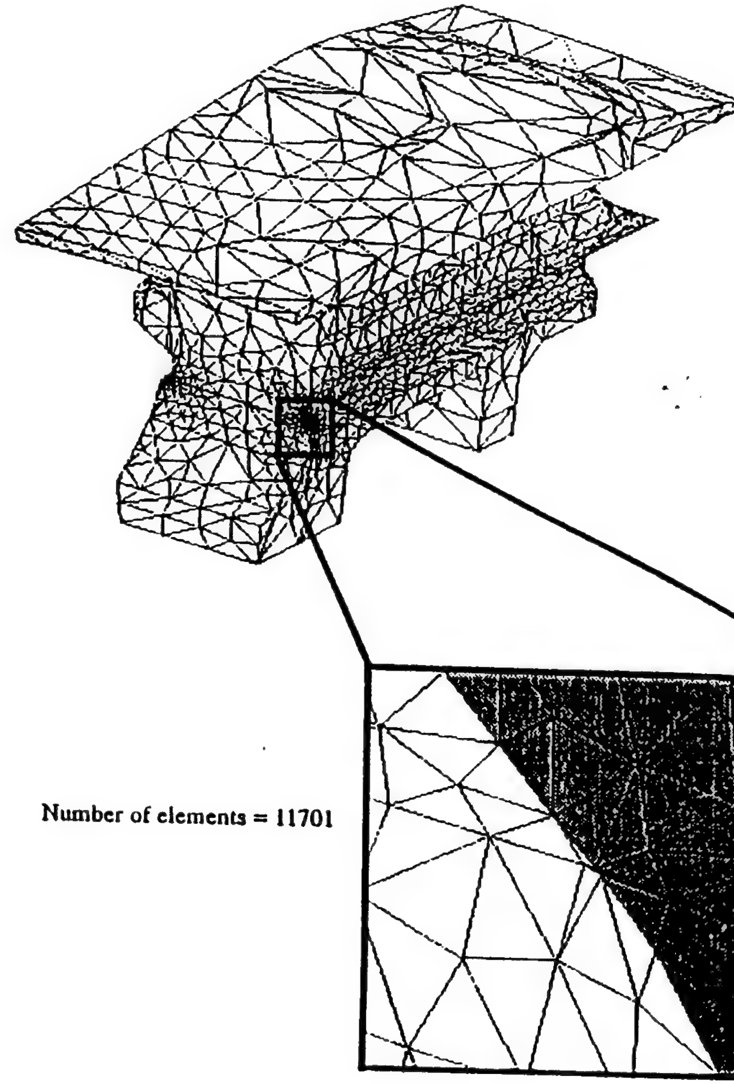
Motivation

- geometric modelers often create very small model features
- excessive mesh refinement is needed to control element quality near small model features
- mesh quality improvement becomes computationally expensive

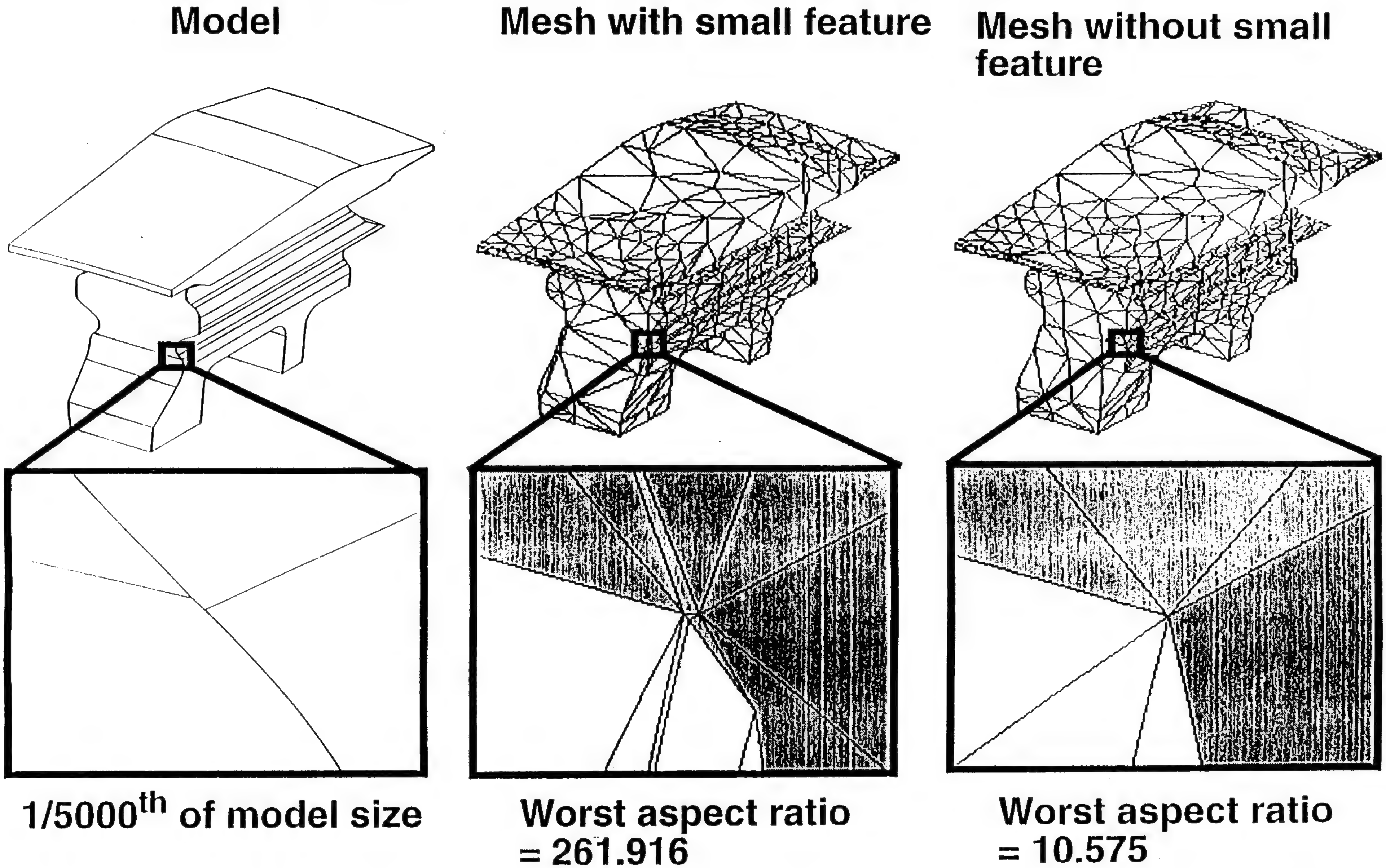
Constraints

- resulting mesh must be acceptable to analysis and downstream procedures; no mechanisms formed
- no loss of analysis attributes information

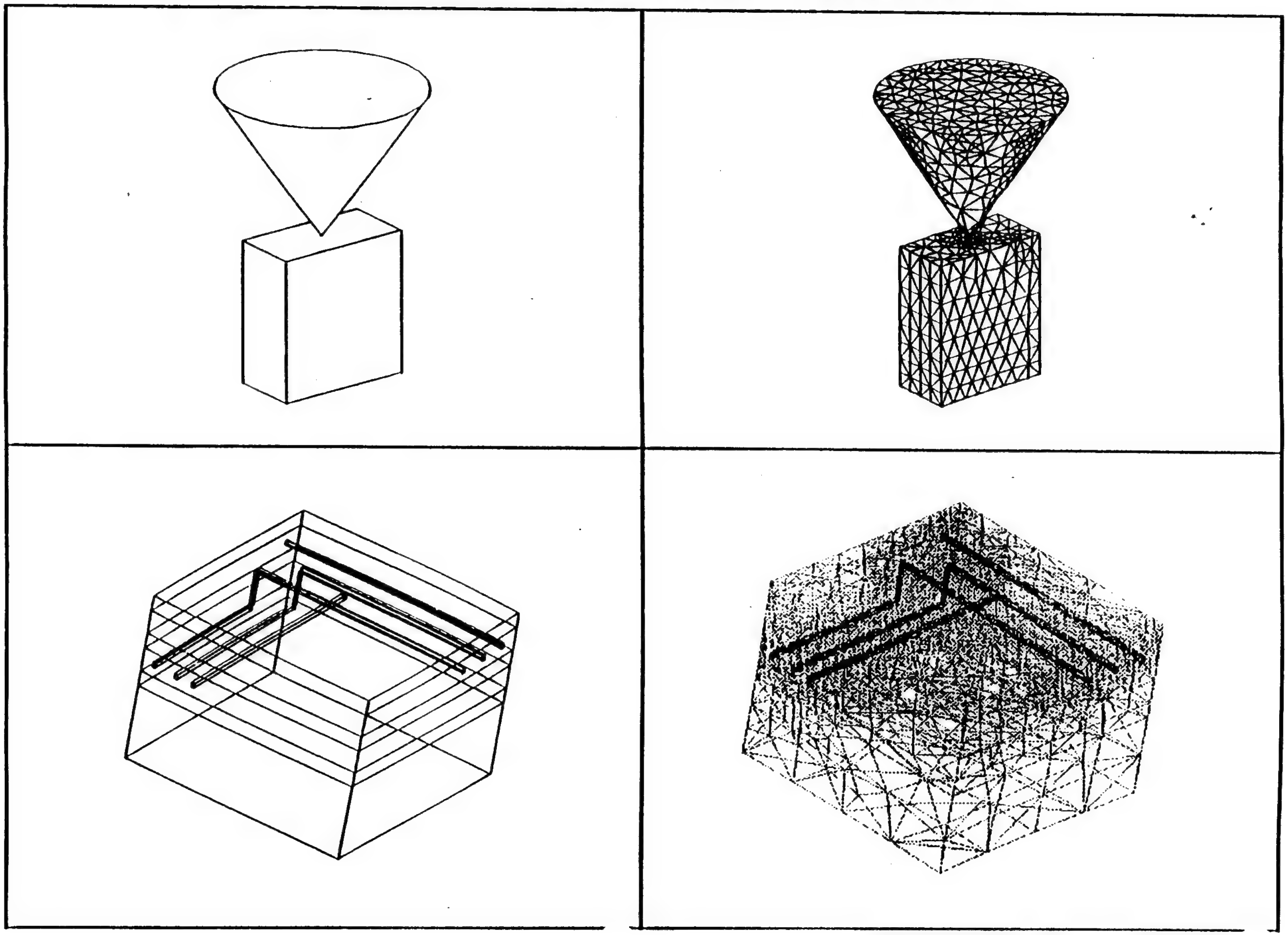

Size of feature
~ 1/5000th model size

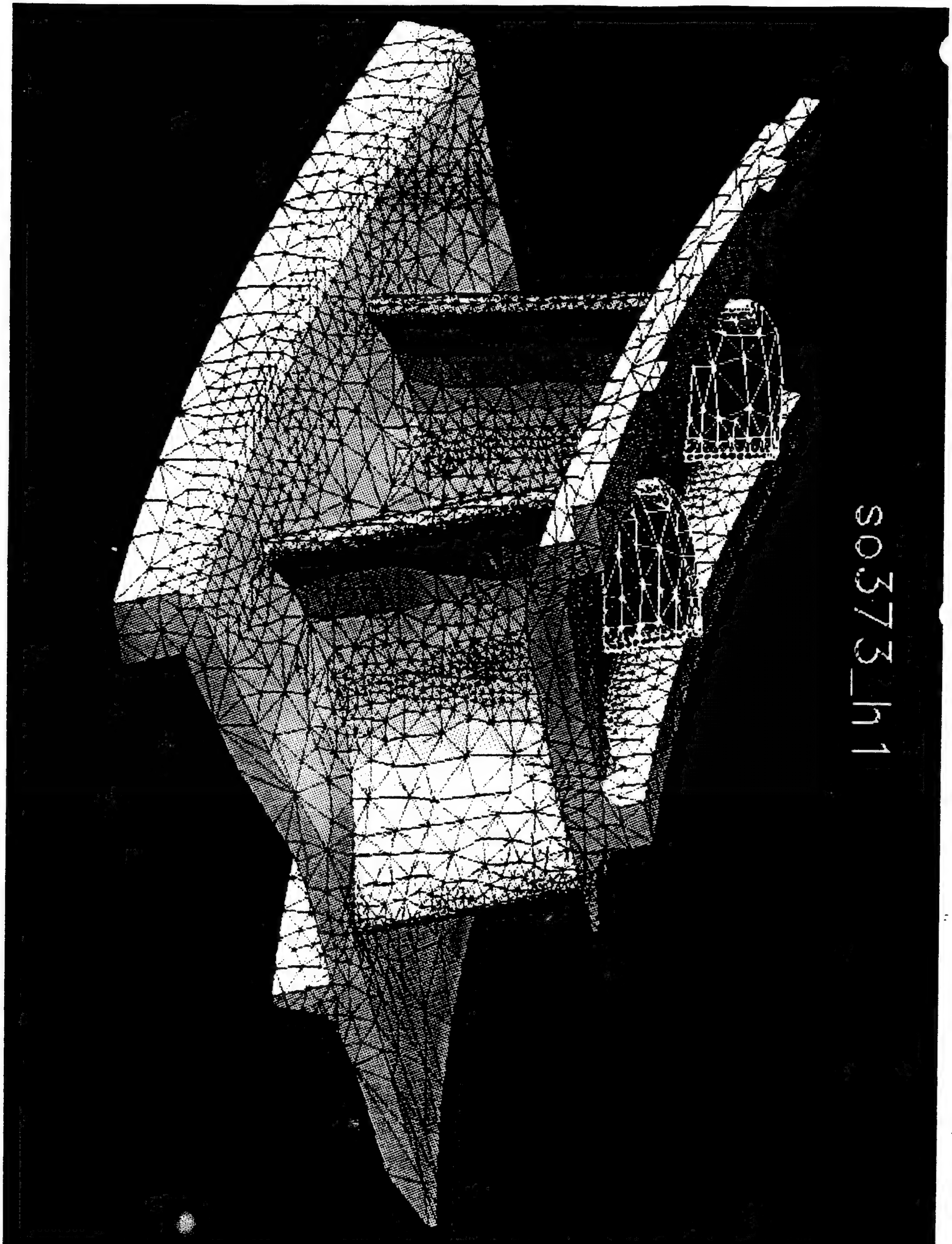

Number of elements = 11701

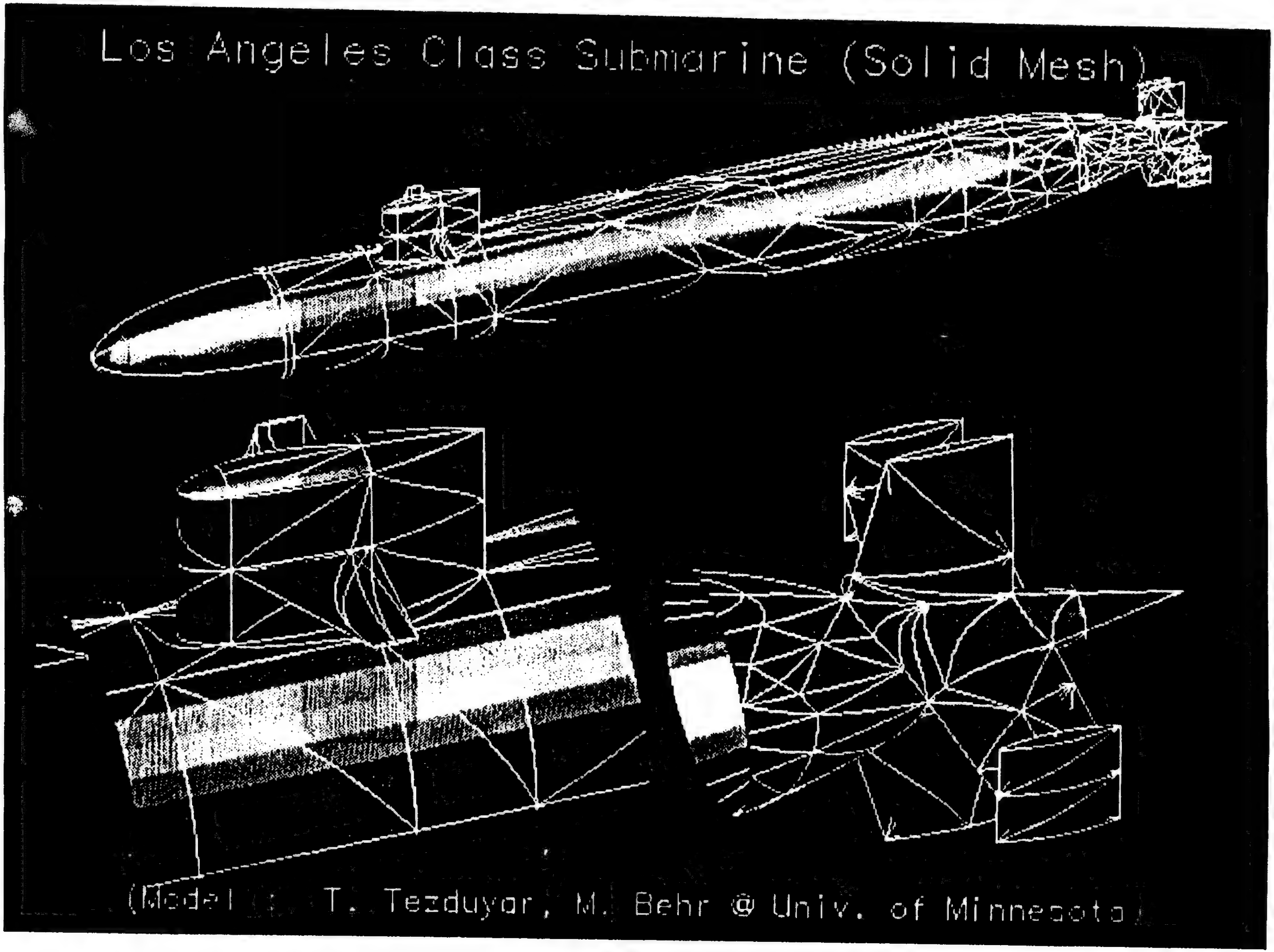
SMALL FEATURE REMOVAL IN FINITE OCTREE

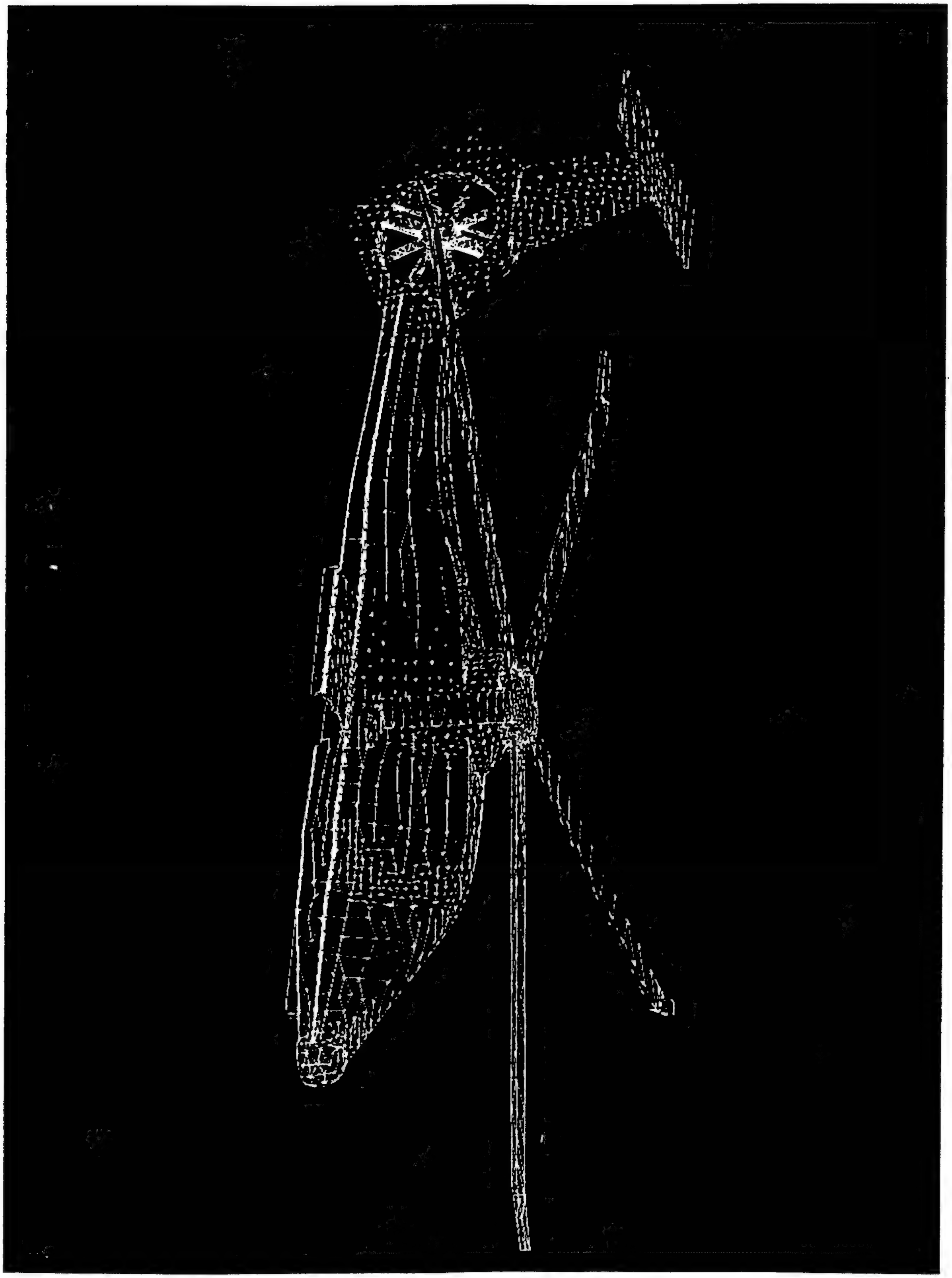


EXAMPLE NON-MANIFOLD MODELS AND THEIR MESHES




S0373_h1

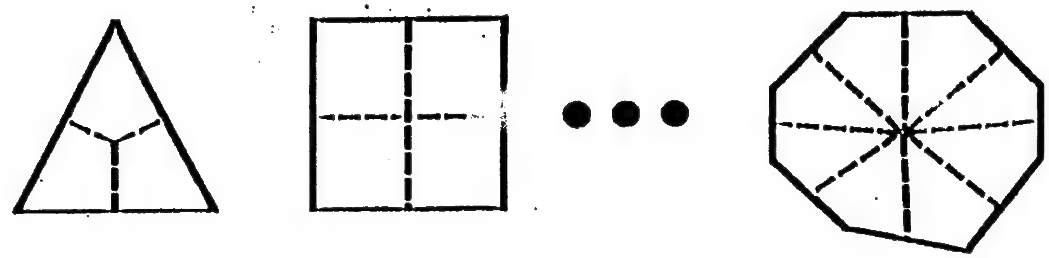

Los Angeles Class Submarine (Solid Mesh)
T. Tezduyar, M. Behr @ Univ. of Minnesota



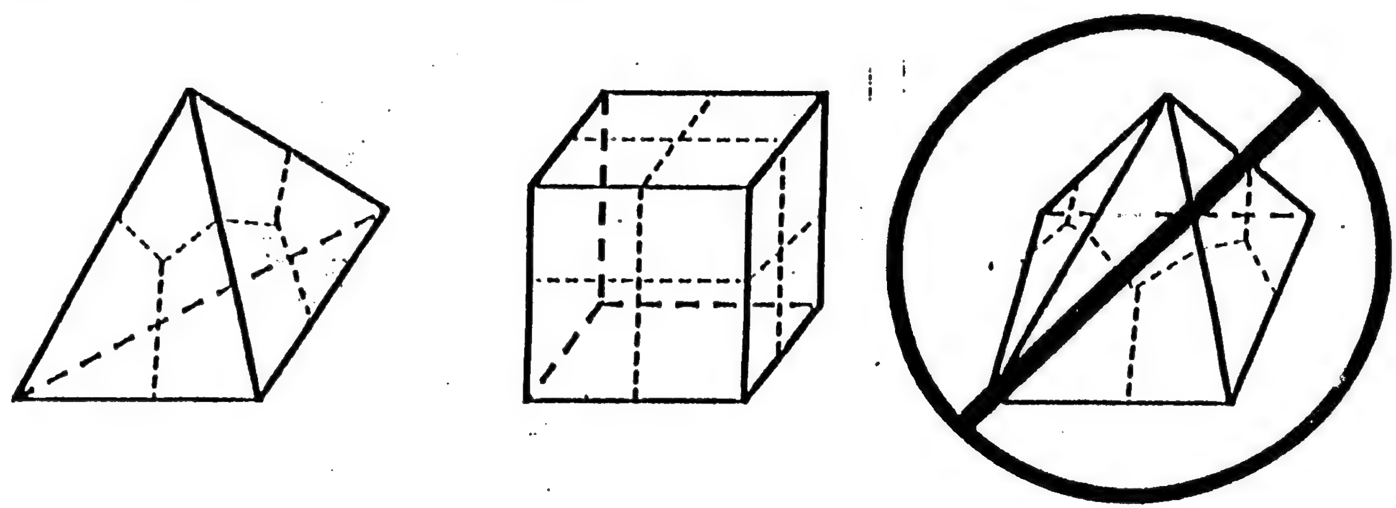
BASIC ISSUE FOR HEXAHEDRAL ELEMENT

Relates to being able to subdivide "simple" shapes and match neighbors

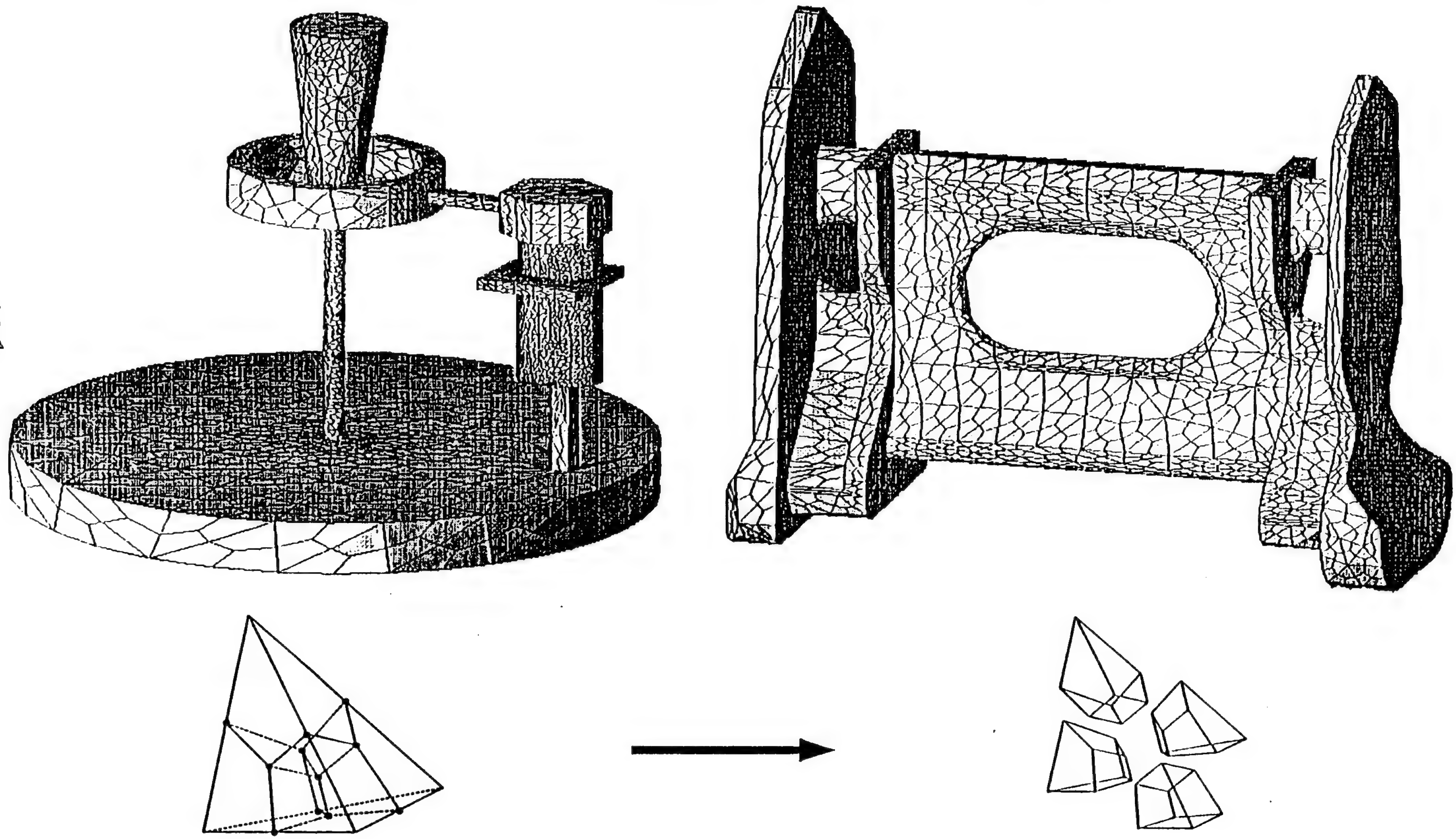
Trivial in 2-D



Pyramid causes problems in 3-D



FINITE OCTREE HEXAHEDRAL SOLID MESHES



FINITE QUADTREE/OCTREE AUTOMATED FEM SYSTEMS

Mesh generators create and control meshes

Tree structure for local control of multiple level refinement and efficient parallel processing

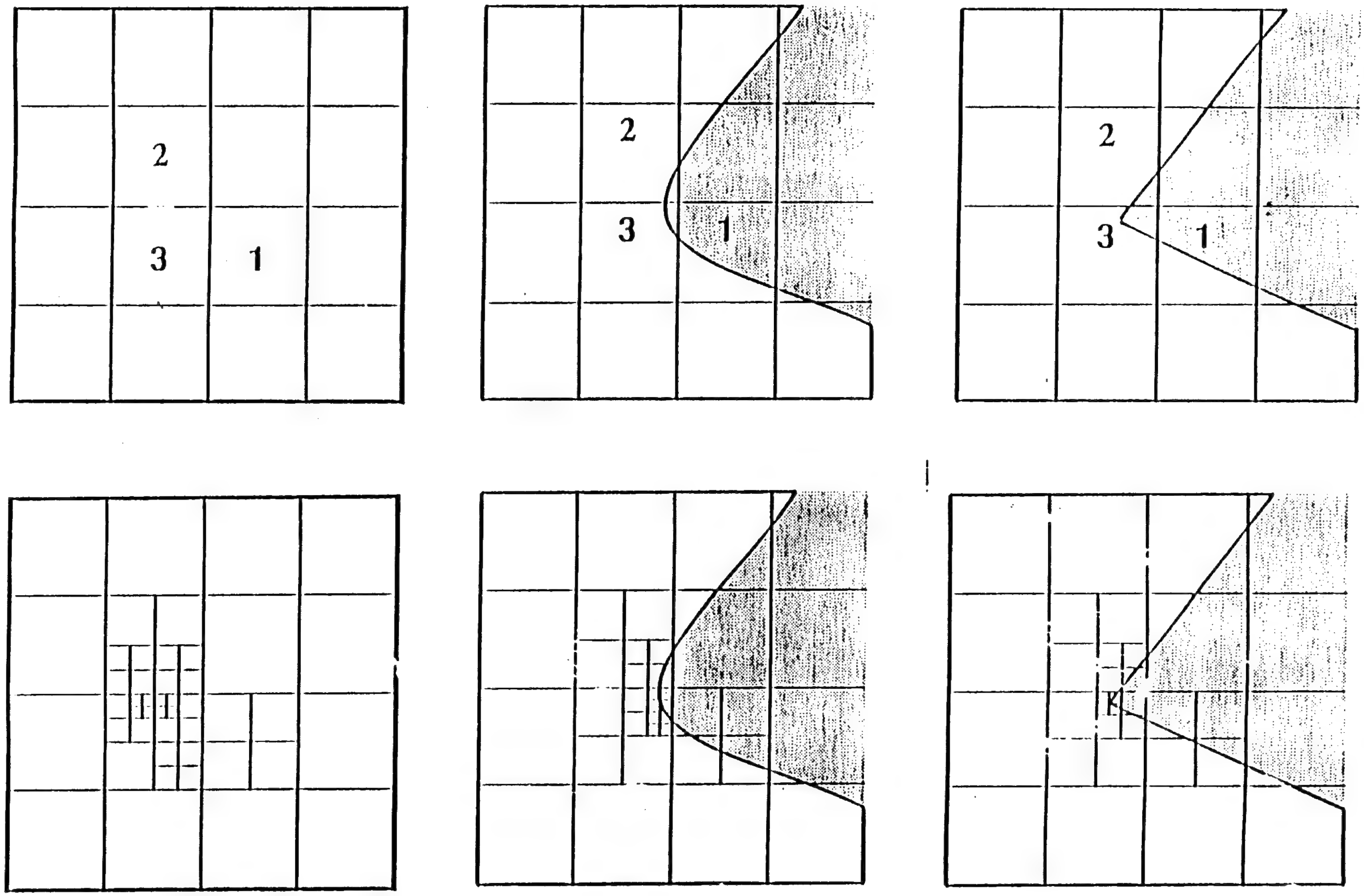
Geometry/mesh links used for evolving geometry

Modular structure allows easy combination with other analysis procedures, error estimators, etc.

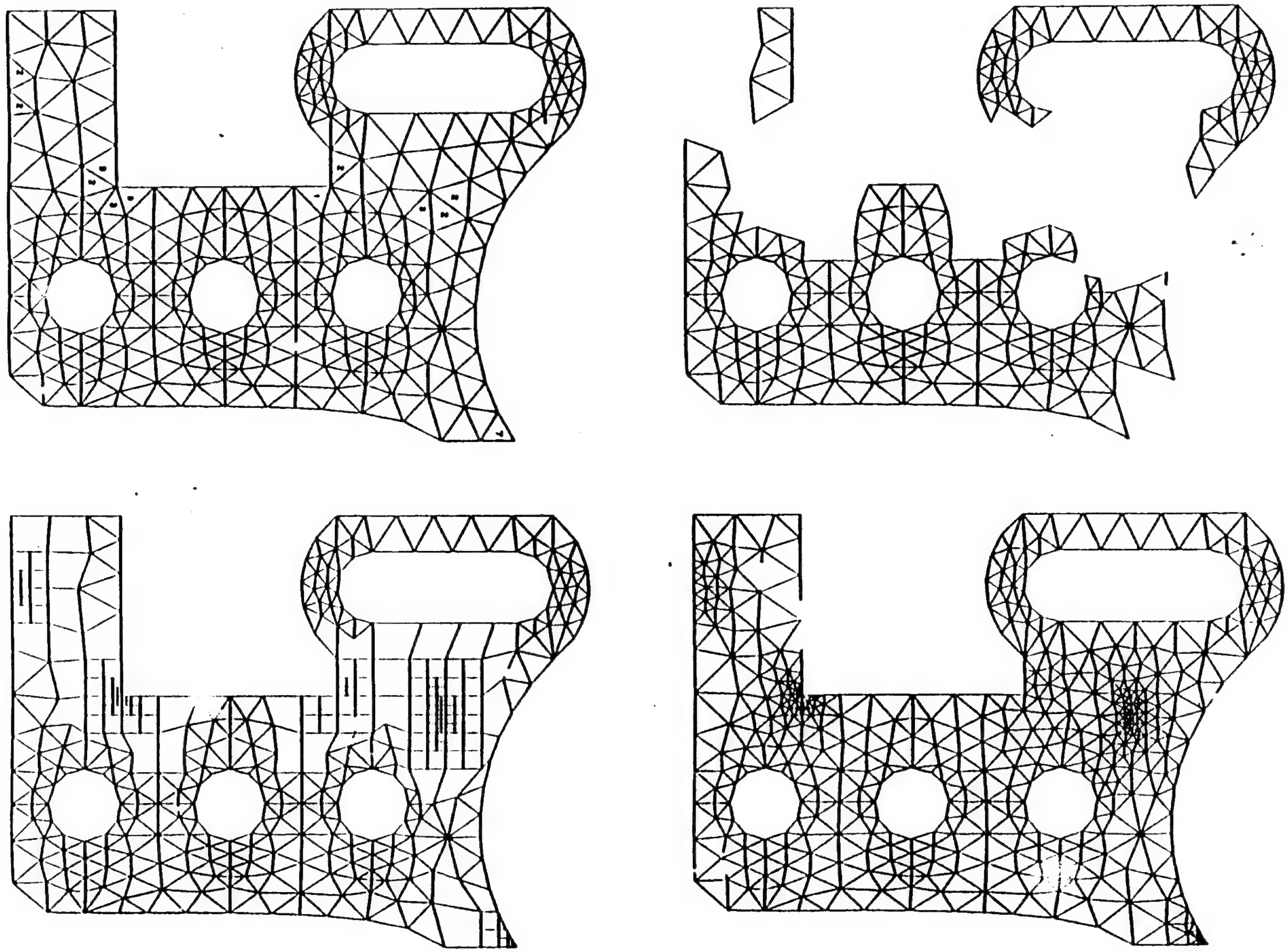
Local remeshing for h-refinement:

- Old element information is deleted
- Quadrants are refined multiple levels in tree, with refinement graded towards boundaries
- Model boundaries are locally discretized
- Finite elements are made in new quadrants and transition zone

Mesh Generation - Local Mesh Updating



Mesh Generation - Local Mesh Updating



RESIDUAL BASED ERROR ESTIMATOR

Babuska has showed that for even order elements the interior residual dominates while for odd order elements the jump term dominates

The assumption of nodal superconvergence of nodal values and ignoring the jumps provides a convenient localization of the error equation for even order elements

For example for u^h and w^h even order polynomials, selecting u^* and w^* in terms of higher order bubble shape functions

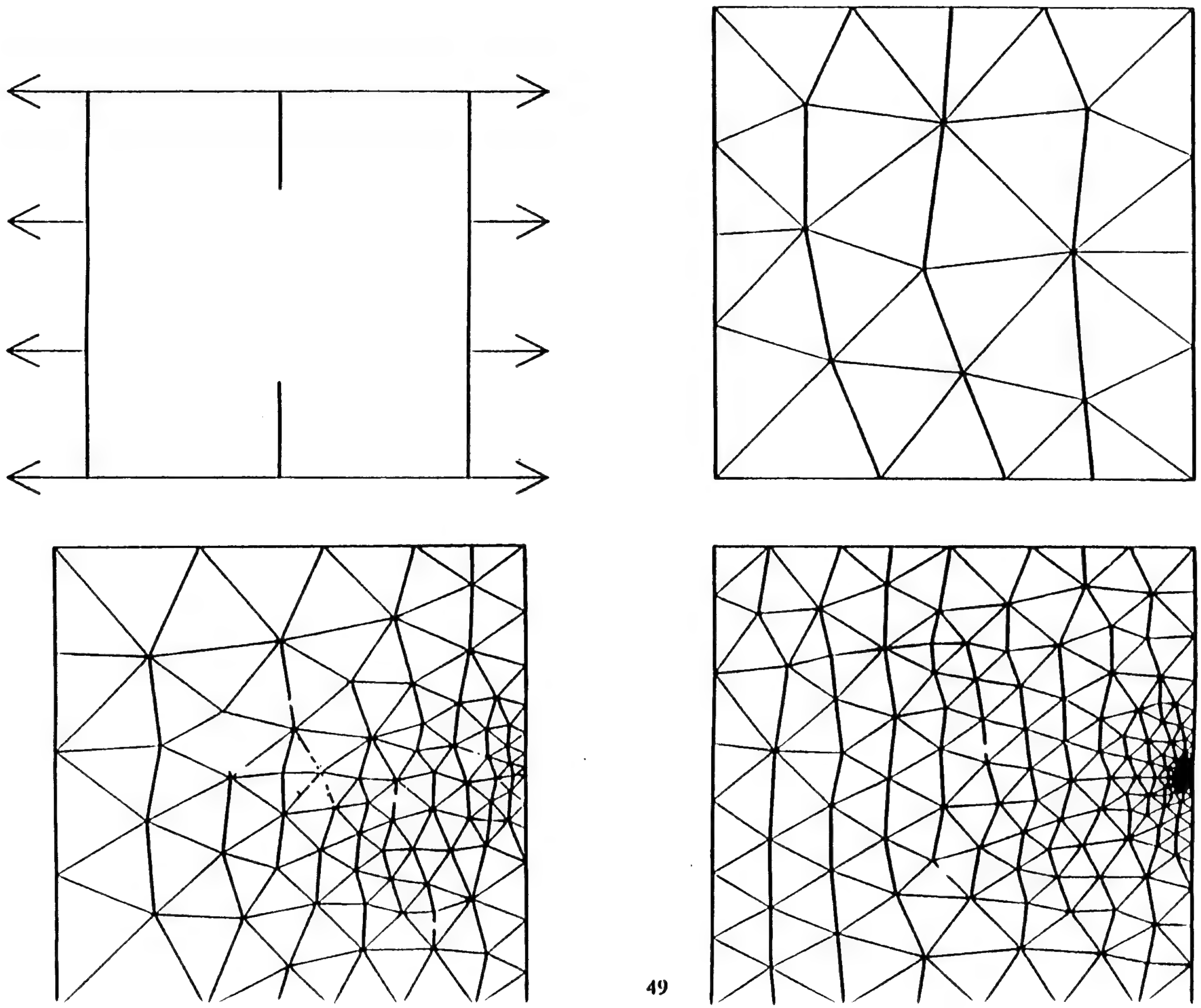
For example on quadratic triangles the shape functions used are

$$N_1^* = \xi_1\xi_2\xi_3 \ , \ N_2^* = \xi_1^2\xi_2\xi_3 \ , \ N_3^* = \xi_1\xi_2^2\xi_3$$

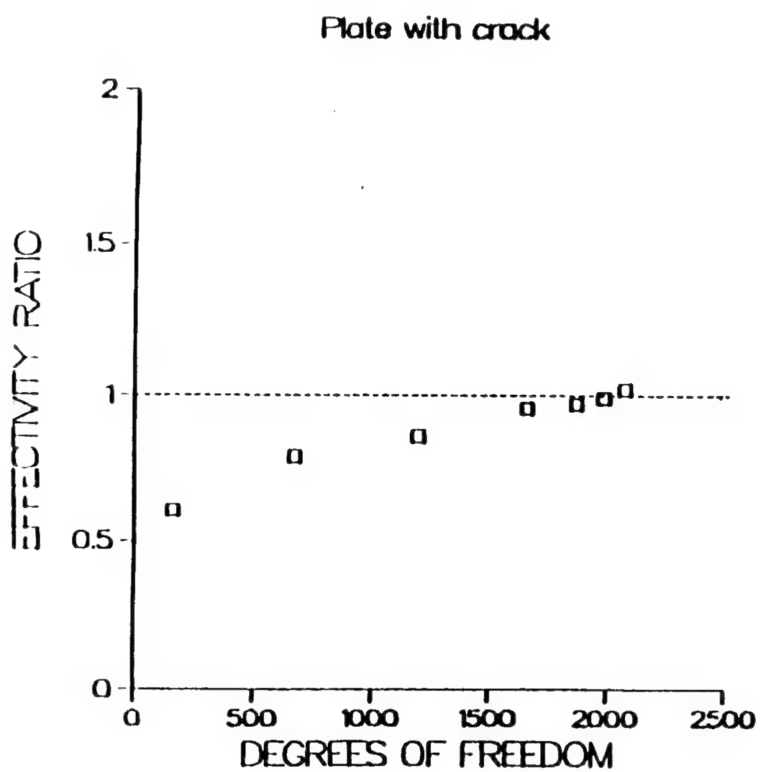
and for quadratic tetrahedra the shape functions used are

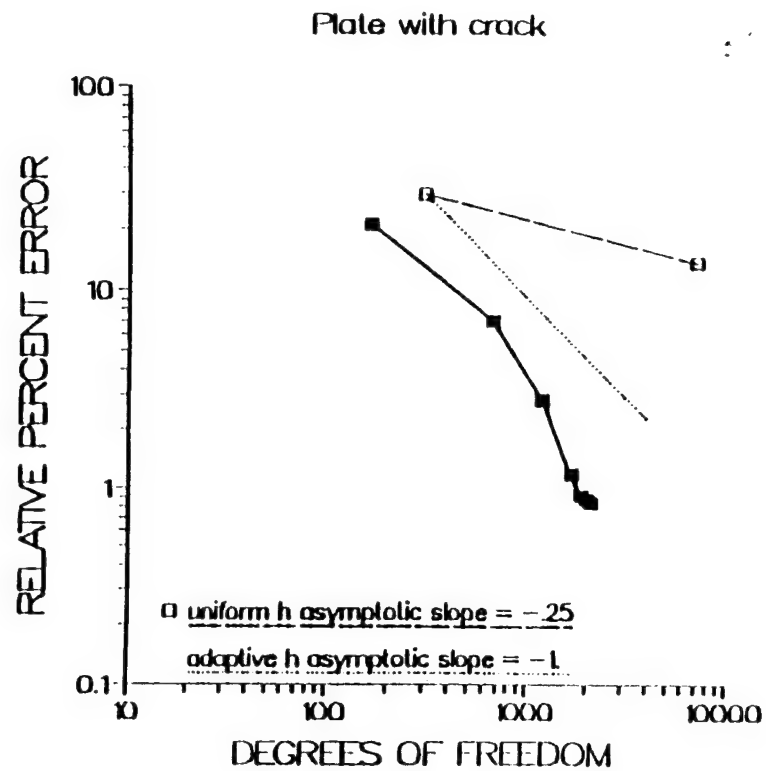
$$N_1^* = \xi_1\xi_2\xi_3\xi_4 \ , \ N_2^* = \xi_1^2\xi_2\xi_3\xi_4 \ , \ N_3^* = \xi_1\xi_2^2\xi_3\xi_4 \ , \ N_4^* = \xi_1\xi_2\xi_3^2\xi_4$$

Adaptive Results – Adaptively Refined Meshes



Adaptive Results – Accuracy





147

BISECTION BASED MESH PREDICTION

- **Utilize mesh topology and classification knowledge**
- **Exploit hierarchy of mesh entities: examine ${}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0}$**
  - **Min/max requested size adjacent to ${}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0}$:**

$$_{\min}\psi_{\mathrm{j}}^{0} = \min_{\left\{{}_{\mathrm{M}}\mathrm{T}_{\mathrm{n}}^{3} \mid {}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0} \subset \partial\left({}_{\mathrm{M}}\mathrm{T}_{\mathrm{n}}^{3}\right)\right\}} \left(\mathrm{h}_{\mathrm{n}}^{\mathrm{new}}\right), \quad _{\max}\psi_{\mathrm{j}}^{0} = \max_{\left\{{}_{\mathrm{M}}\mathrm{T}_{\mathrm{n}}^{3} \mid {}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0} \subset \partial\left({}_{\mathrm{M}}\mathrm{T}_{\mathrm{n}}^{3}\right)\right\}} \left(\mathrm{h}_{\mathrm{n}}^{\mathrm{new}}\right)$$

  - **If ${}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0} \sqsubset {}_{\mathrm{G}}\mathrm{T}_{\mathrm{k}}^{0}$, possible error source: bisect until**

$$\mathrm{h}_{\mathrm{i}} = {}_{\min}\psi_{\mathrm{j}}^{0}, \ \forall \left\{{}_{\mathrm{M}}\mathrm{T}_{\mathrm{i}}^{3} \mid {}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0} \subset \partial\left({}_{\mathrm{M}}\mathrm{T}_{\mathrm{i}}^{3}\right)\right\}$$

  - **Else limit excessive refinement: bisect until**

$$\mathrm{h}_{\mathrm{i}} = {}_{\max}\psi_{\mathrm{j}}^{0}, \ \forall \left\{{}_{\mathrm{M}}\mathrm{T}_{\mathrm{i}}^{3} \mid {}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0} \subset \partial\left({}_{\mathrm{M}}\mathrm{T}_{\mathrm{i}}^{3}\right)\right\}$$

  - **Coarsen ${}_{\mathrm{M}}\mathrm{T}_{\mathrm{k}}^{3}$ if** $\mathrm{h}_{\mathrm{k}}^{\mathrm{old}} < \left({}_{\max}\psi_{\mathrm{j}}^{0}, {}_{\min}\psi_{\mathrm{j}}^{0}\right), \ \left\{{}_{\mathrm{M}}\mathrm{T}_{\mathrm{k}}^{3} \mid {}_{\mathrm{M}}\mathrm{T}_{\mathrm{j}}^{0} \subset \partial\left({}_{\mathrm{M}}\mathrm{T}_{\mathrm{k}}^{3}\right)\right\}$
  - **Control abrupt h gradation via further bisections**

BISECTION BASED MESH PREDICTION (CONT.)

- Examine ${}_{M}T_j^1$ to further define distribution
  - If ${}_{M}T_j^1 \sqsubset {}_{G}T_k^1$ and $\partial\left({}_{M}T_j^1\right) \not\sqsubset {}_{G}T_n^0$, ${}_{G}T_k^1$ possible error source: bisect until $h_i = {}_{\min}\psi_j^1,\ \forall \left\{ {}_{M}T_i^3 \mid {}_{M}T_m^1 \subset \partial\left({}_{M}T_i^3\right) \right\}$ where ${}_{M}T_m^1$ = subdivision of ${}_{M}T_j^1$
  - If ${}_{\min}\psi_j^1 < \min\limits_{\left\{ {}_{M}T_m^0 \mid {}_{M}T_m^0 \subset \partial\left({}_{M}T_j^1\right) \right\}} \left({}_{\min}\psi_m^0\right)$, error source in edge neighborhood: bisect until $h_i = {}_{\min}\psi_j^1$, for some $\left\{ {}_{M}T_i^3 \mid {}_{M}T_m^1 \subset \partial\left({}_{M}T_i^3\right) \right\}$
  - If $\partial\left({}_{M}T_j^1\right) \sqsubset {}_{G}T_n^0$, ${}_{G}T_n^0$ possible error source: bisection along ${}_{M}T_j^1$ unlikely to improve accuracy
  - Else bisect until $h_i = {}_{\max}\psi_j^1,\ \forall \left\{ {}_{M}T_i^3 \mid {}_{M}T_m^1 \subset \partial\left({}_{M}T_i^3\right) \right\}$
- Coarsen ${}_{M}T_k^3$ if $h_k^{old} < \left({}_{\max}\psi_j^1, {}_{\min}\psi_j^1\right)$, $\left\{ {}_{M}T_k^3 \mid {}_{M}T_m^1 \subset \partial\left({}_{M}T_k^3\right) \right\}$
- Repeat with ${}_{M}T_j^2$, ${}_{M}T_j^3$ (check for $\partial\left({}_{M}T_j^d\right) \sqsubset {}_{G}T_n^{d-1}$) to complete definition

IMPLEMENTATION OF MULTIPLE LEVEL REFINEMENT

- **Loop over mesh entities in spatial order** $d = 0, \ldots, 3$
- **Determine max new size requested adjacent to** ${}_M T_i^d$**:**

$$\psi_i^d = \max_{\{{}_M T_n^3 | {}_M T_i^d \subset \partial({}_M T_n^3)\}} (h_n^{new})$$

- **Determine** $\min$ **of** ψ **values for** $\left\{{}_M T_j^{d-1} | {}_M T_j^{d-1} \subset \partial({}_M T_i^d)\right\}$**:**

$$_{min/max}\psi_i^d = \min\left(\psi_I^{d-1}, \ldots, \psi_\gamma^{d-1}\right)$$

- **If** ${}_M T_i^d \sqsubset {}_G T_j^d$ **and** $\partial({}_M T_i^d) \not\sqsubset {}_G T_n^{d-1}$ **assign** ${}_M T_i^d$ **the value**

$$_{min}\psi_i^d = \min_{\{{}_M T_n^3 | {}_M T_i^d \subset \partial({}_M T_n^3)\}} (h_n^{new})$$

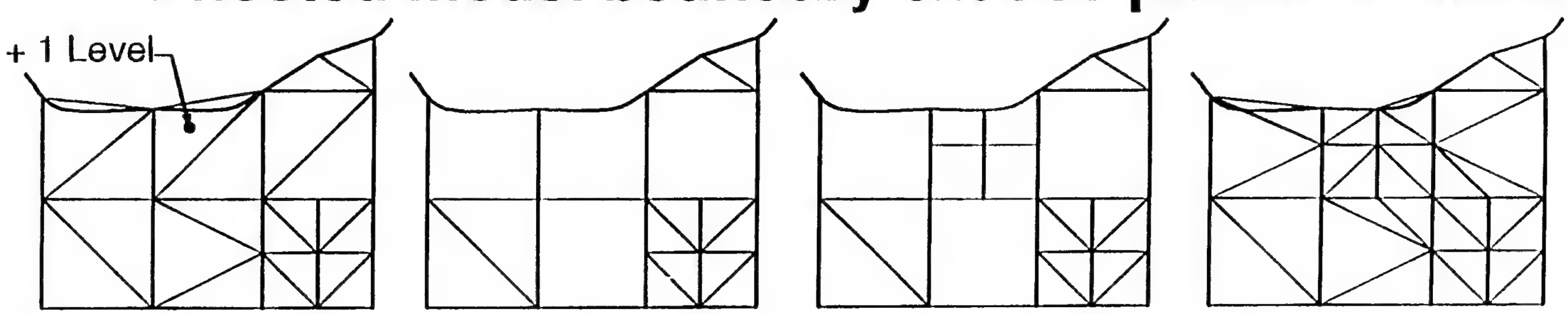
- **If** ${}_M T_i^d \not\sqsubset {}_G T_j^d$ **assign** ${}_M T_i^d$ **the value** $_{min/max}\psi_i^d$
- **If** $\psi_i^d < {}_{min/max}\psi_i^d$ **assign additional value** ψ_i^d **to** ${}_M T_i^d$
- **Note:** $h_n^{new} > h_n^{old} \rightarrow$ **coarsening**

MULTIPLE LEVEL IMPLEMENTATION (CONT.)

- Specify distribution to local remesh as refine points
  - One point for vertices
  - Uniform (parametric) points for higher entities
- Calculate required tree levels at refine points

$$\text{element size} = (\text{universe size}) * \left(\frac{1}{2}\right)^{\text{level}}$$

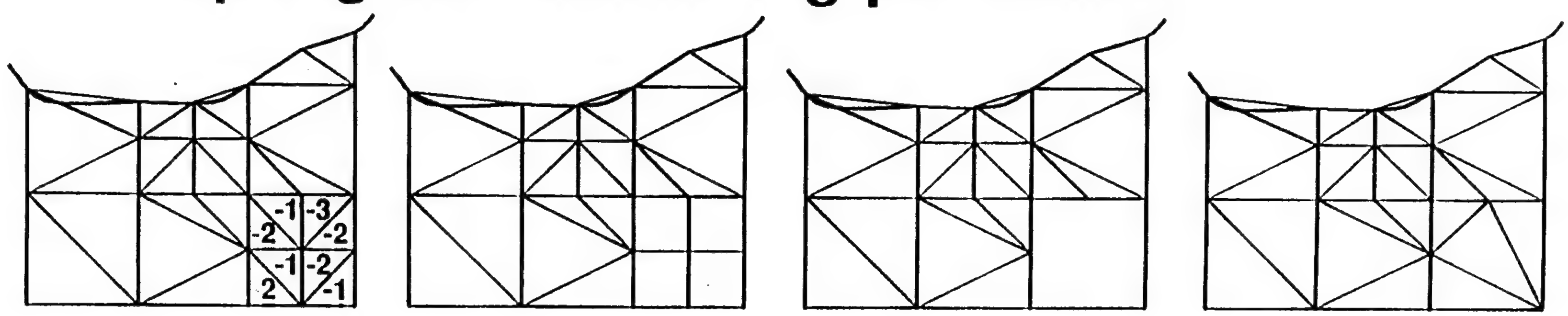
- All octants containing point refined as requested
- If octant requires refinement:
  - Contained mesh entities deleted
  - Mesh entities in face and edge neighbors deleted
  - Effected model boundary entities placed in stack



148

MULTIPLE LEVEL IMPLEMENTATION (CONT.)

- Octants requiring coarsening are coarsened if:
  - Not previously refined
  - Siblings require at least same level of coarsening
  - Parent and neighbors meet one level difference
- If the octant can be coarsened:
  - Effected mesh entities deleted
  - Effected model boundary entities placed in stack
- Necessary model entities reinserted
- Elements formed in areas of tree modification
- Collapsing and smoothing performed



EXAMPLES

- Examine convergence (error in energy norm)
  - Asymptotic rate of h-adaptive convergence:

$$||\mathrm{u} - \mathrm{U}|| \leq \mathrm{K}\mathrm{N}^{-\frac{1}{2}\mathrm{p}}$$

$||\mathrm{u} - \mathrm{U}|| =$ error in energy norm

$\mathrm{N} =$ total degrees of freedom

$\mathrm{K} =$ constant, $\mathrm{p} =$ element basis order

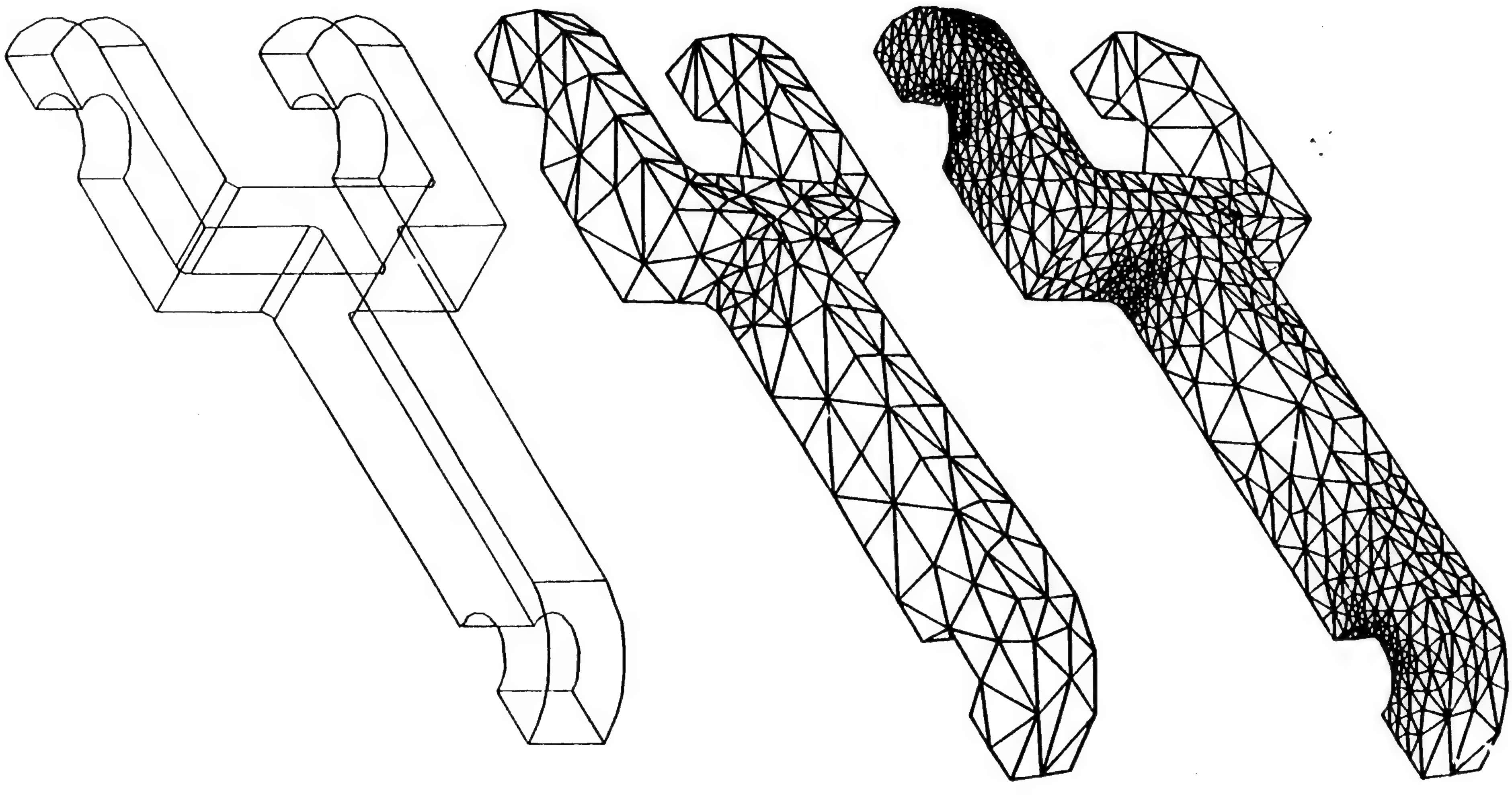
  - $\log(||\mathrm{u} - \mathrm{U}||/||\mathrm{u}|| \times 100\%))$ / $\log(\mathrm{N}) \rightarrow \mathrm{slope} = -1$ $(\mathrm{p} = 2)$
- Evaluate error estimator quality (effectivity ratio Θ)
  - If exact solution available: $\Theta = ||\mathrm{E}||/||\mathrm{u} - \mathrm{U}||$
  - If exact solution unavailable:

$$\Theta = ||\mathrm{E}||/||\overline{\mathrm{u}} - \mathrm{U}||, \quad ||\overline{\mathrm{u}}|| = ||\mathrm{U}_{\mathrm{finest}} + \mathrm{E}_{\mathrm{finest}}||$$

  - Exact error predicted $\rightarrow \Theta = 1$
- SUN/SPARC 10,128Mbytes real,1.7Gbytes swap/temp

150

3–D ADAPTIVE RESULTS — FLANGE EXAMPLE



CONCLUSIONS

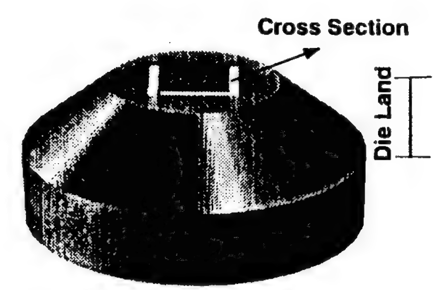
- Developed automated h-adaptive analysis with PATRAN 2.5 and commercial modeler
- Examples illustrate most time spent in analysis

Operation	Percent of elapsed time
Generation of all meshes	4.96% - 7.65%
Completion of all analyses	74.85% - 82.31%

- Final analysis most expensive: 63.9% - 79.92%
- Initial mesh, analysis, error est.: 3.75% - 11.25%
  - Targets areas for refinement → greater accuracy
  - Avoids overly fine initial mesh → faster analyses
- Efficiency improved if singularity degree carefully approximated

ALUMINIUM EXTRUSION DIE SPECIFICATION

- Extrusion of Aluminium is done using flat-faced dies
- There are four types of flat-faced dies used for Aluminium extrusion
  - solid-shape
  - porthole
  - bridge
  - baffle or feeder-plate
- Solid-shape dies are primarily used for extruding solid shapes like T section, L section etc.



solid-shape die used for Aluminium extrusion

MODEL PARAMETERS FOR SOLID-SHAPE DIES

- The die models have been defined using the AML virtual geometric modeler
- The main parameters which control the solid-shape of the die are:
  - Diameter of the die
  - Diameter of the work piece
  - Type of cross section to be extruded
  - Length of die land
- The generated models are non-manifold with two different material regions:
  - One material region for the work piece
  - One material region for the die
  - Complete knowledge of the interface maintained

MODEL DEFINITION WITH AML VIRTUAL GEOMETRIC MODELER

- Process of defining die models has been automated by writing a generalized AML input file
- The input file has been parametrized interms of
  - Diameters of the die and the work piece
  - Length of die land
  - Cross section to be extruded
- Different die shapes can be generated by changing the parameters in this file
- Output from AML is the model meshed by Finite Octree
- Finite Octree queries the modeler to obtain the necessary information about the model

MESHING DIE MODELS USING FINITE OCTREE

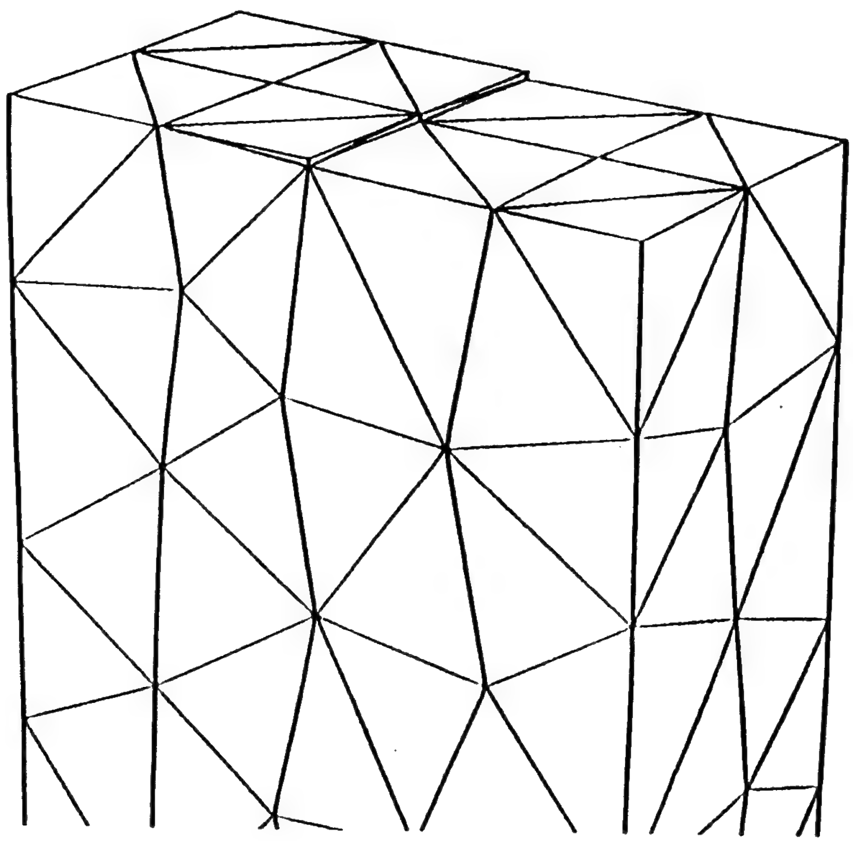
- Finite Octree mesh generator has been used to generate the mesh for the die models
- Finite Octree supports a wide range of methods to control element sizes and mesh gradations
- The different mesh controls available are
  - Specification of element sizes
  - Curvature dependent refinement
  - Detection of small model features and control of element aspect ratios
- Mesh control parameters should be assigned to topological entities
- Element size specification and curvature based refinement has been used to generate the example meshes

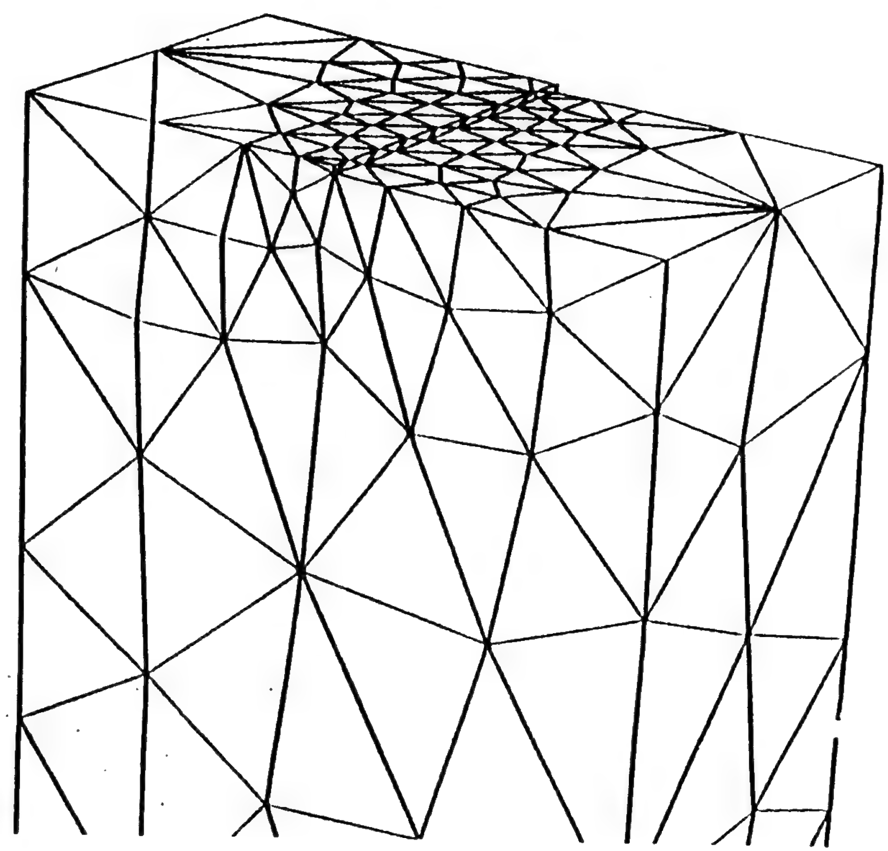
ELEMENT SIZE SPECIFICATION BASED ON MODEL ENTITIES

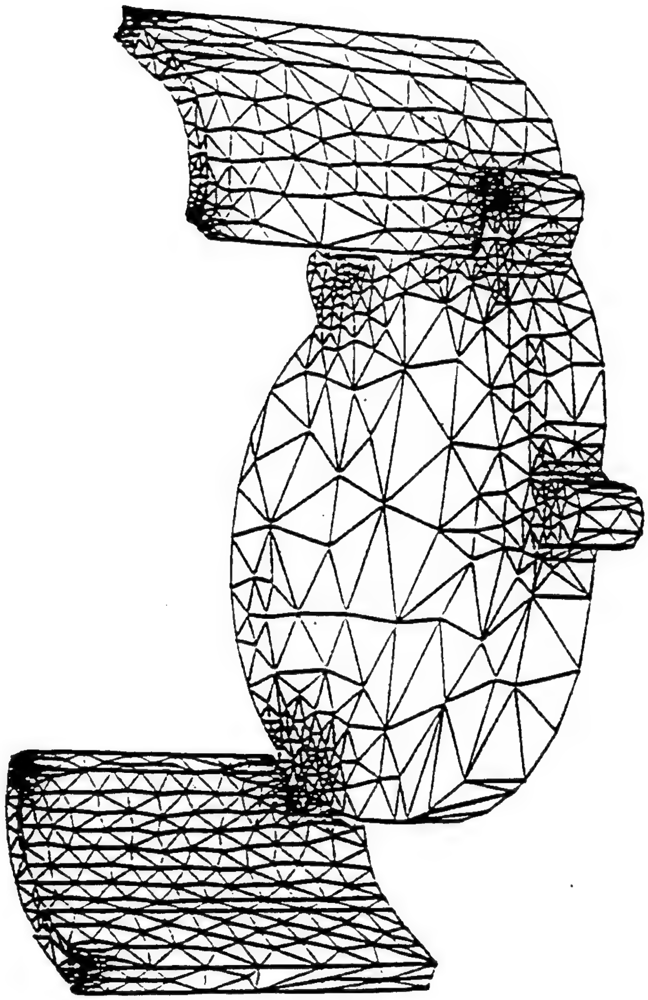
Specify element size associated with model vertices, edges faces and/or regions

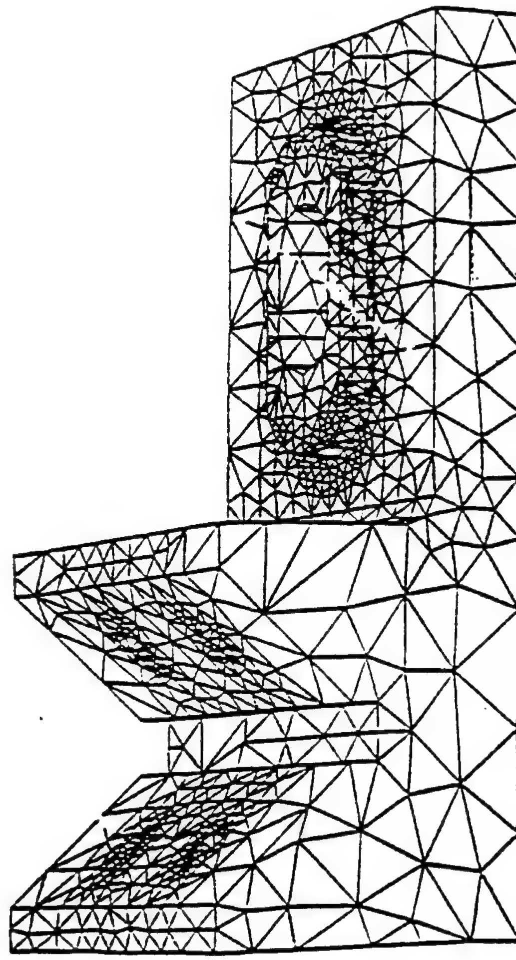
Smallest size overrides

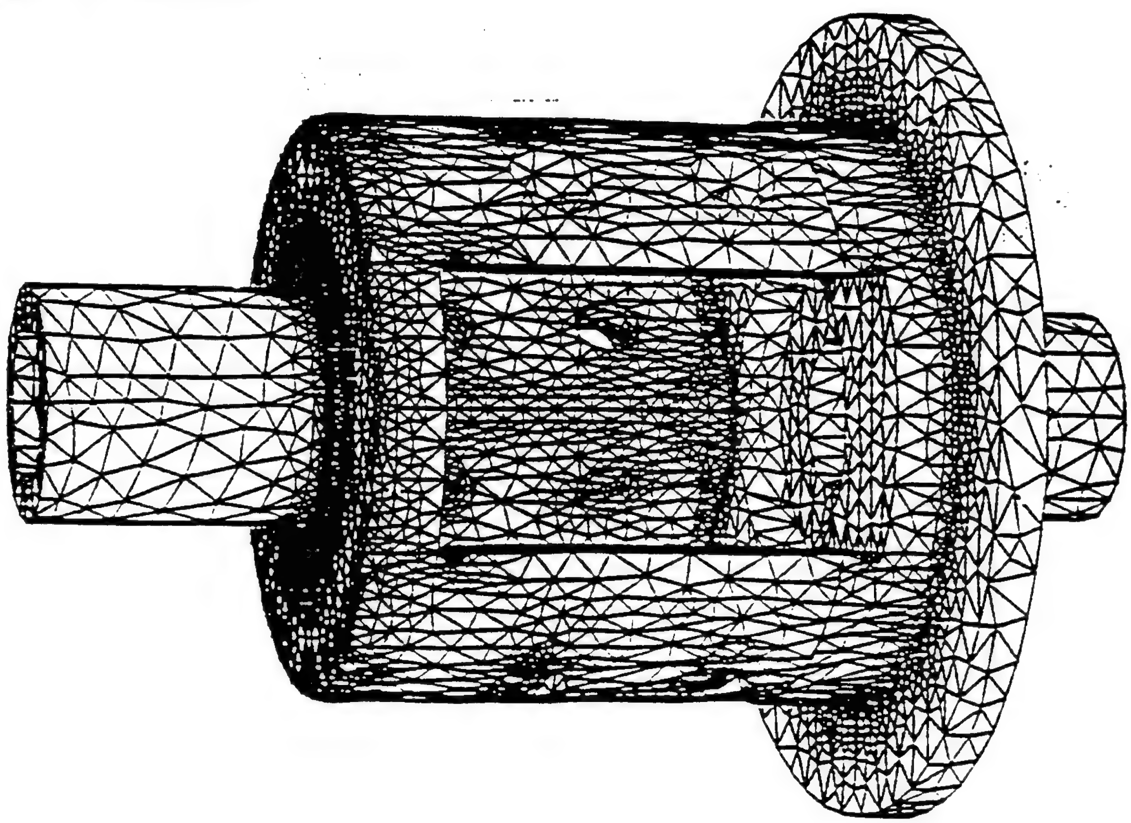
Mesh generator responsible for creating gradations





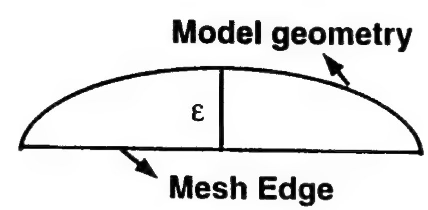






DESCRIPTION OF THE MESH CONTROL PARAMETERS

- Mesh control can be applied globally or locally (individual mesh entities)
- Element sizes are specified as desired length of mesh edges
- Curvature based refinement controls the approximation of the curved geometric entities
- Controlled by the geometric approximation factor ϵ
- Geometric approximation is defined as the maximum distance between the mesh entity and the portion of the model it approximates



Geometric approximation factor

MESH CONTROL PARAMETERS

- Element sizes are specified by giving a minimum and maximum required element sizes
- Two different methods available for specifying geometric approximation

1. Constant geometric approximation for model entity
   - —Approximates geometry equally over model entity
   - —Specified as explicit distance or fraction of model size
   - —Well suited for constant curvature surfaces, e.g., cylinder, spheres
2. Geometric approximation as fraction of element size
   - —Approximation varies with curvature of surface
   - —Well suited for surfaces with variable curvatures, e.g., free from surfaces

160

MESH CONTROL PARAMETERS FOR T DIE AND MESH STATISTICS

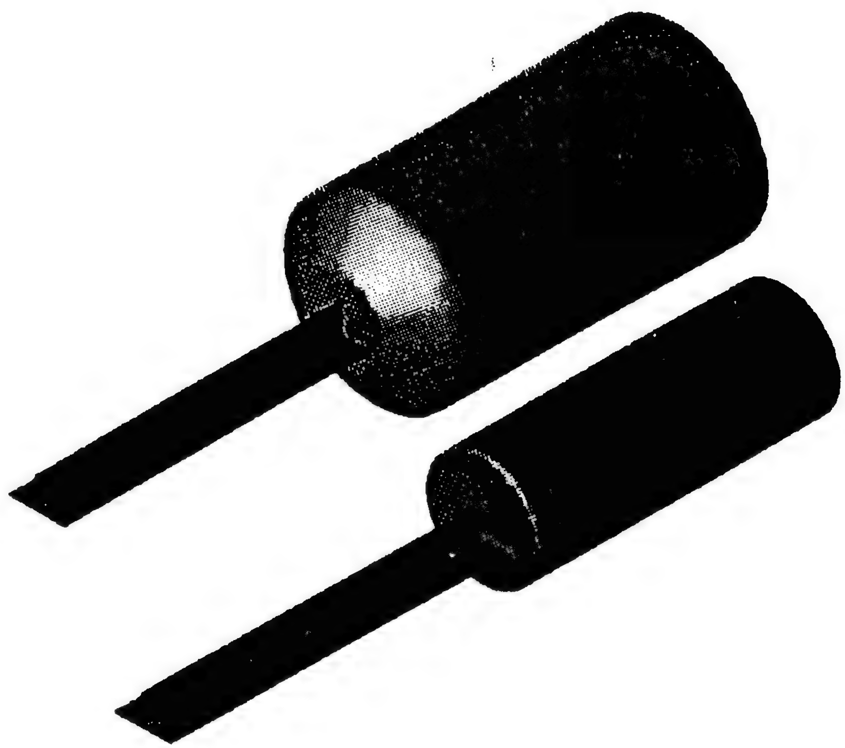
Mesh control parameters

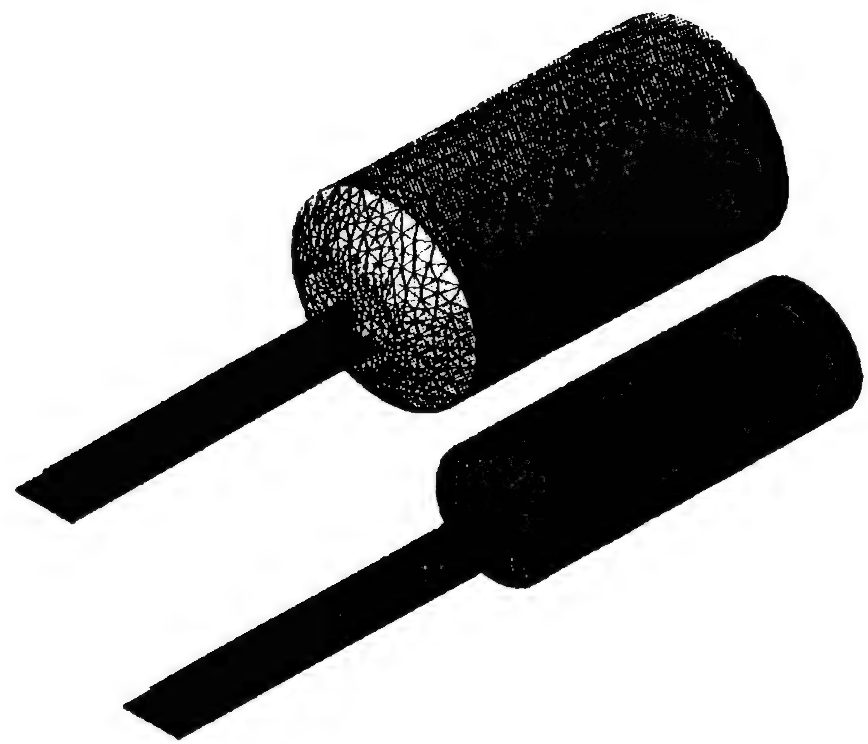
Minimum element size (global)	0.0453
Maximum element size (global)	0.7261
Curvature refinement method (global)	1
Maximum allowable geometric approximation (global)	0 5
Maximum element size at the die fluid interaction (local)	0.18154

Mesh statistics

Number of nodes	4675
Number of surface triangles	4582
Number of tetrahedrons	23770
Smallest dihedral angle	4.594
Largest dihedral angle	163.42
Largest edge/shortest height	21.892

MODEL AND MESH FOR T SECTION





MESH CONTROL PARAMETERS FOR L DIE AND MESH STATISTICS

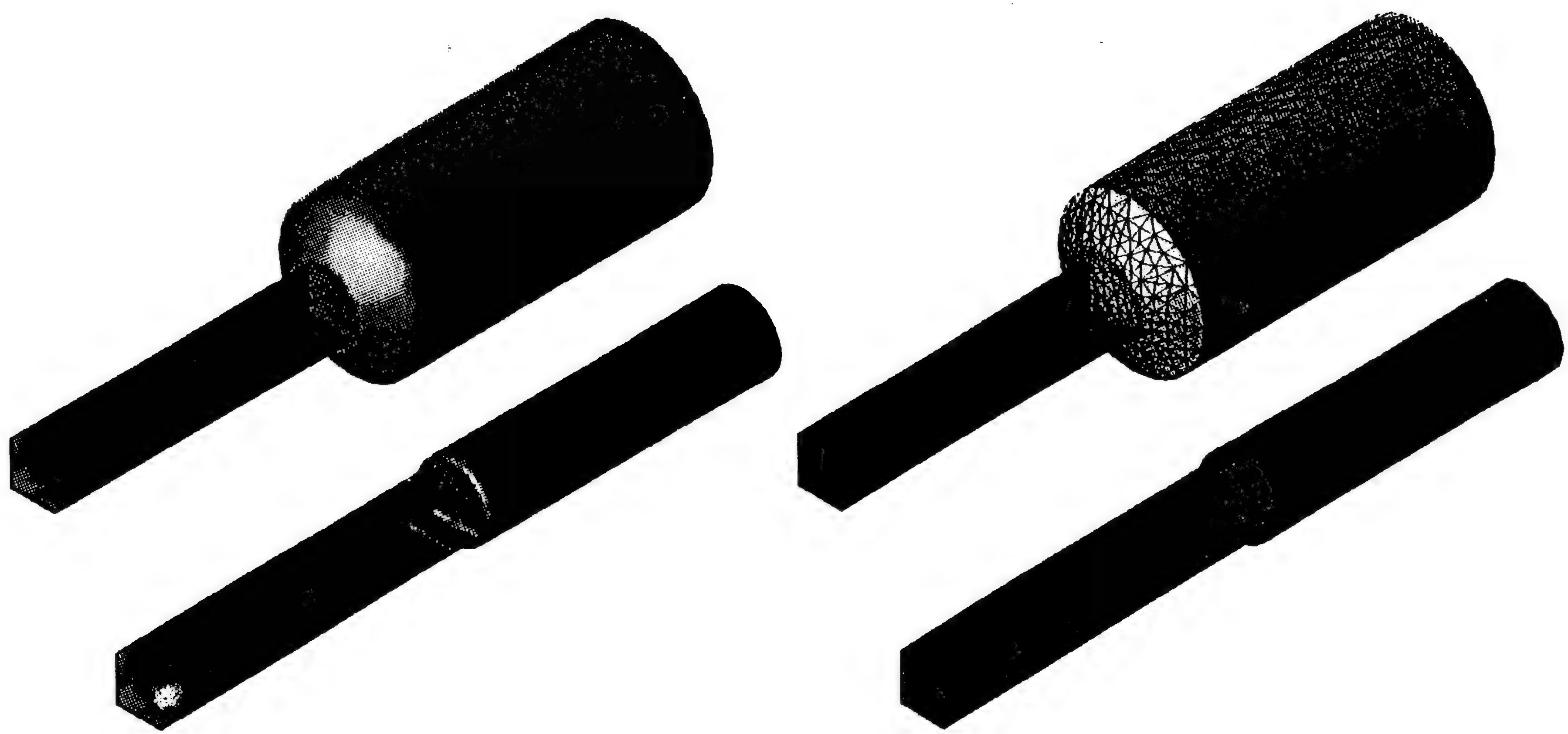
Mesh control parameters

Minimum element size (global)	0.04490
Maximum element size (global)	0.7184
Curvature refinement method (global)	1
Maximum allowable geometric approximation (global)	0.5
Maximum element size at the die fluid interaction (local)	0.17960

Mesh statistics

Number of nodes	3283
Number of surface triangles	3376
Number of tertrahedrons	16052
Smallest dihedral angle	9.969
Largest dihedral angle	148.455
Largest edge/shortest height	10.760

MODEL AND MESH FOR L SECTION



MESH CONTROL PARAMETERS FOR CHANNEL DIE AND MESH STATISTICS

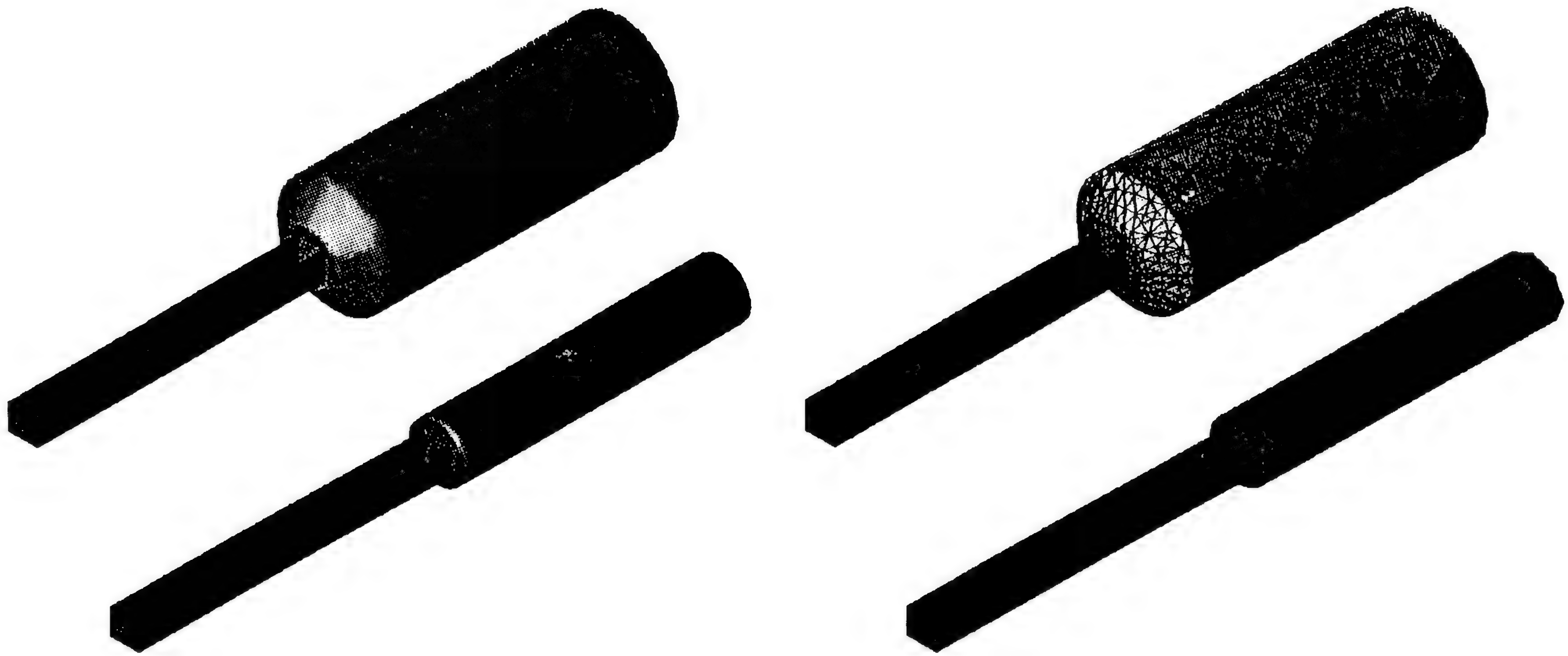
Mesh control parameters

Minimum element size (global)	0.04485
Maximum element size (global)	0.7166
Curvature refinement method (global)	1
Maximum allowable geometric approximation (global)	0.3
Maximum element size at the die fluid interaction (local)	0.17940

Mesh statistics

Number of nodes	3073
Number of surface triangles	3546
Number of tertrahedrons	14992
Smallest dihedral angle	5.046
Largest dihedral angle	153.164
Largest edge/shortest height	19.436

MODEL AND MESH FOR CHANNEL SECTION



INPUT TO ANALYSIS PROCEDURE

- Finite Octree generates a generic neutral file for the generated mesh along with the classification information
- Analysis procedure (Spectrum) needs following input
  - Node numbers and the x,y,z coordinates of the nodes
  - Connectivity for the tetrahedrons (set of four nodes)
  - The node numbers on the inflow surface on the die and the work piece
  - The list of surface triangles (set of three nodes) lying at the fluid solid interface, along with the tetrahedron on the solid and fluid side
  - The list of surface triangles (set of three nodes) lying on the free surface of the die

SOFTWARE DEVELOPED FOR WRITING SPECTRUM INPUT FILE

- Interface developed to write the Spectrum input file
- The classification information associated with the mesh entities used to derive the required information
- The SCOREC mesh database operators used to
  - Specify boundary triangle information
  - Get the classification information for the mesh entities
  - Get the connectivity information for the mesh entities
- Node numbers are assigned to all the mesh vertices
- The connectivity information returned by the mesh database operators are converted to node numbers and written to Spectrum input file

1168

TETRAHEDRAL MESH GENERATION WITH MULTIPLE ELEMENTS THROUGH THE THICKNESS

Rao Garimella, Mark S. Shephard
Scientific Computation Research Center
Rensselaer Polytechnic Institute, Troy NY 12180

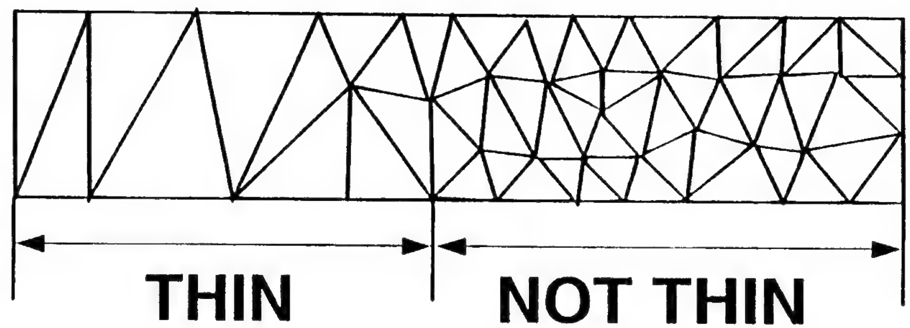
- Generation of directionally stretched elements
- Useful in simulations where gradients are much stronger along some directions than others
- Uniform refinement -> substantial increase in computational cost
- Applications:
  - Multiple elements through thin sections
  - Boundary layer generation

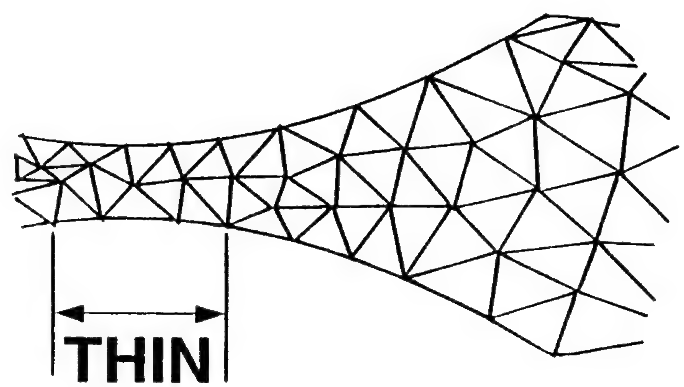
Multiple elements through thin sections

- Define "thin" assuming model discretization available
- *Path*: ordered set of mesh edges connecting 2 vertices
- *Path length*: Number of edges in path
- Model defined to be *locally thin* at given mesh vertex on model face, if shortest path to another (or same) model face is smaller than of user defined length
- Restrictions:
  - shortest path cannot lie entirely on starting face
  - if path is between same model face, it must be shorter than shortest boundary path
- *Boundary path*: path in which all edges classified on model boundary

ILLUSTRATION OF THIN SECTIONS

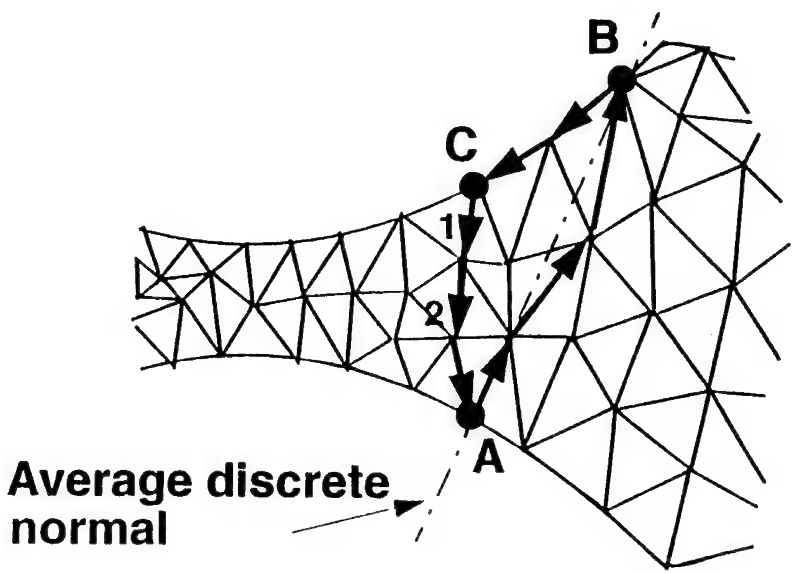
Assume 3 elements requested through thickness





DETECTION OF LOCALLY THIN SECTIONS

1. <u>Forward search</u> from start vertex to find another vertex classified on model face
2. <u>Boundary search</u> on opposite model face to find closest vertex to start vertex - define as *opposite* vertex
3. <u>Reverse search</u> to find shortest path between opposite and start vertices.



A: Start vertex

B: Forward search result

C: Boundary search result

C12A: Shortest path from C to A

A & C are *opposite vertices*

172

DETECTION OF THIN SECTIONS (CONTD.)

Assume N_t elements requested through thickness

Forward search:

- Start at mesh vertex classified on closure of model face
- Calculate average discrete normal of model face at vertex
- Pick mesh edge of vertex most aligned with normal and go to its other vertex
- Search ends when mesh vertex on model face found or (N_t-1) edges have been traversed
- Constant computational complexity for given N_t

Boundary Search:

- Start at mesh vertex found on forward search
- Go to adjacent mesh vertex classified on model face that is closest to start vertex

DETECTION OF THIN SECTIONS (CONTD.)

- Repeat until local miminum reached
- Constant computational cost with limited iterations

Reverse search:

- Start at vertex found on boundary search
- Use greedy shortest path algorithm to find target vertex (start vertex of forward search)
- Go to adjacent vertex that minimizes distance to target vertex
- Repeat until target vertex is reached
- Constant computational cost for given N_t

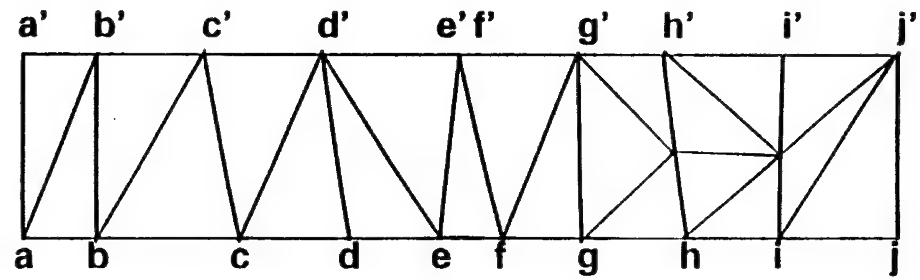
Assuming N_b is the number of boundary mesh vertices, total computational cost of finding opposite vertices is $O(N_b)$

CREATING MULTIPLE ELEMENTS THROUGH THICKNESS

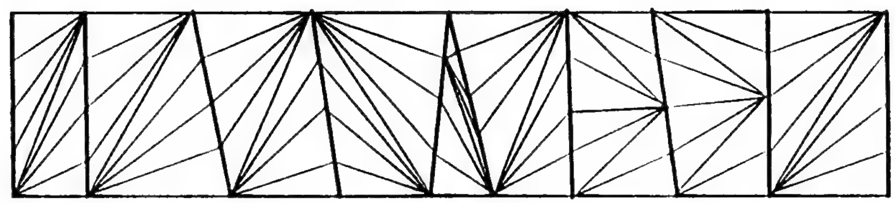
- Create by local mesh modification (edge split, swap) procedures made as efficient possible for this application
- Steps:

1. For each model face
   - Find mesh vertices on closure of model face
   - Find opposite vertices
   - Match mesh on opposite face(s) by edge swapping
   - Split edges between opposite vertices as required
   - Swap edges between opposite edges to get proper alignment of elements
2. Post-process mesh to improve quality

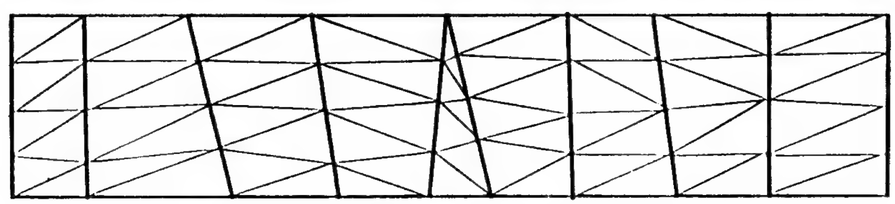
CREATING MULTIPLE ELEMENTS THROUGH THICKNESS (CONTD.) - 2D ILLUSTRATION



Initial mesh with opposite vertices identified

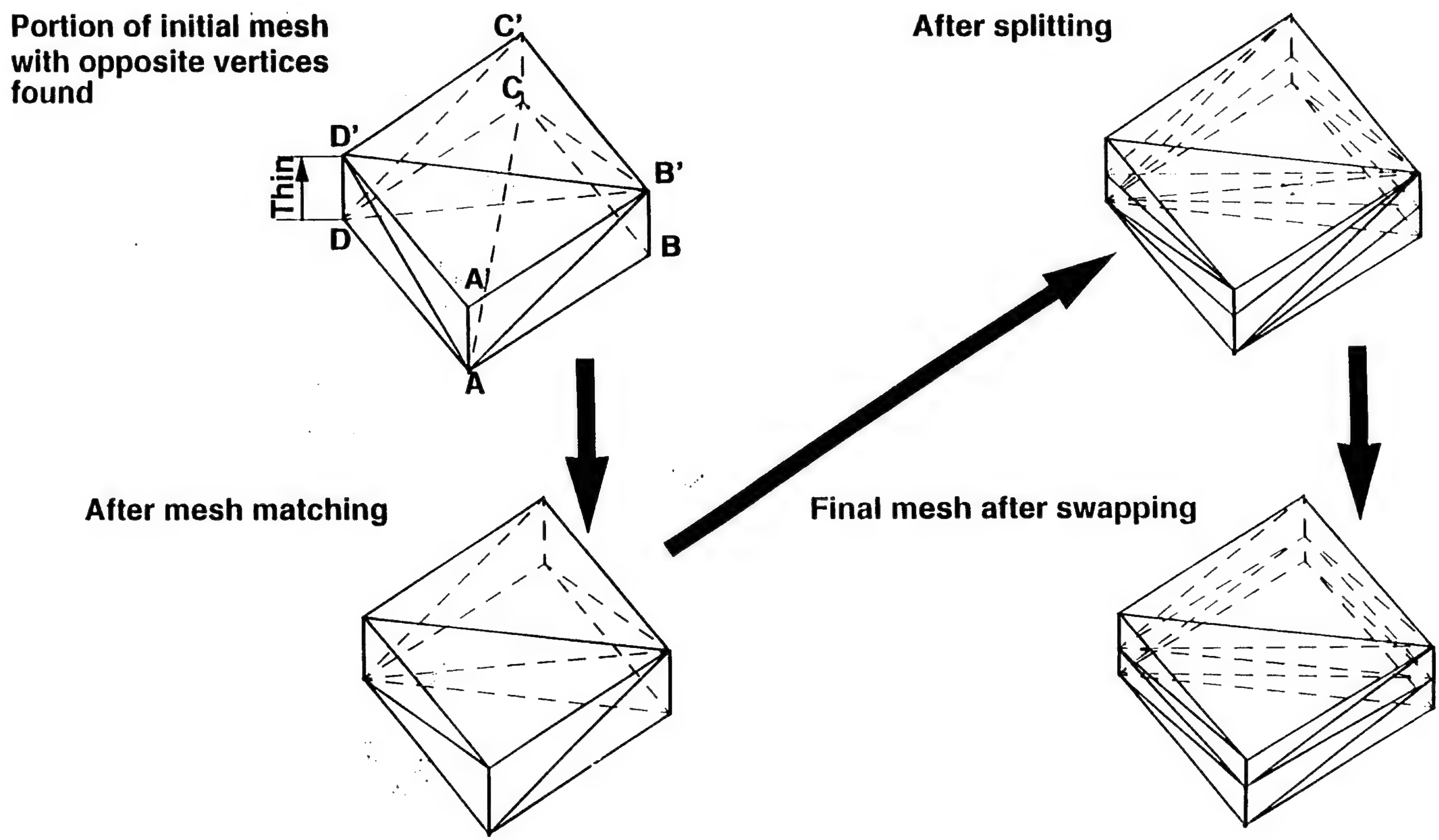


Mesh after splitting



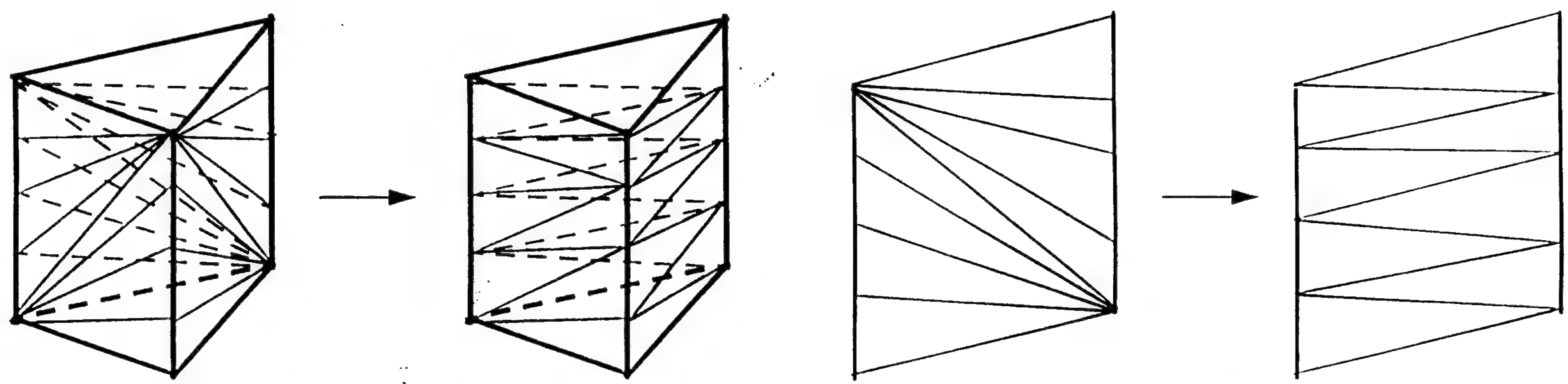
Final mesh after swapping

CREATING MULTIPLE ELEMENTS THROUGH THICKNESS (CONTD.) - 3D ILLUSTRATION



SWAPPING EDGES

- Swapping of edges after splitting to
  - align edges along and perpendicular to thickness
  - evenly redistribute connections (to decrease bandwidth of stiffness matrix)
- Consider groups of elements as wedges through the thickness with triangulated faces
- Triangulation on faces may be "diagonal" or "zigzag"

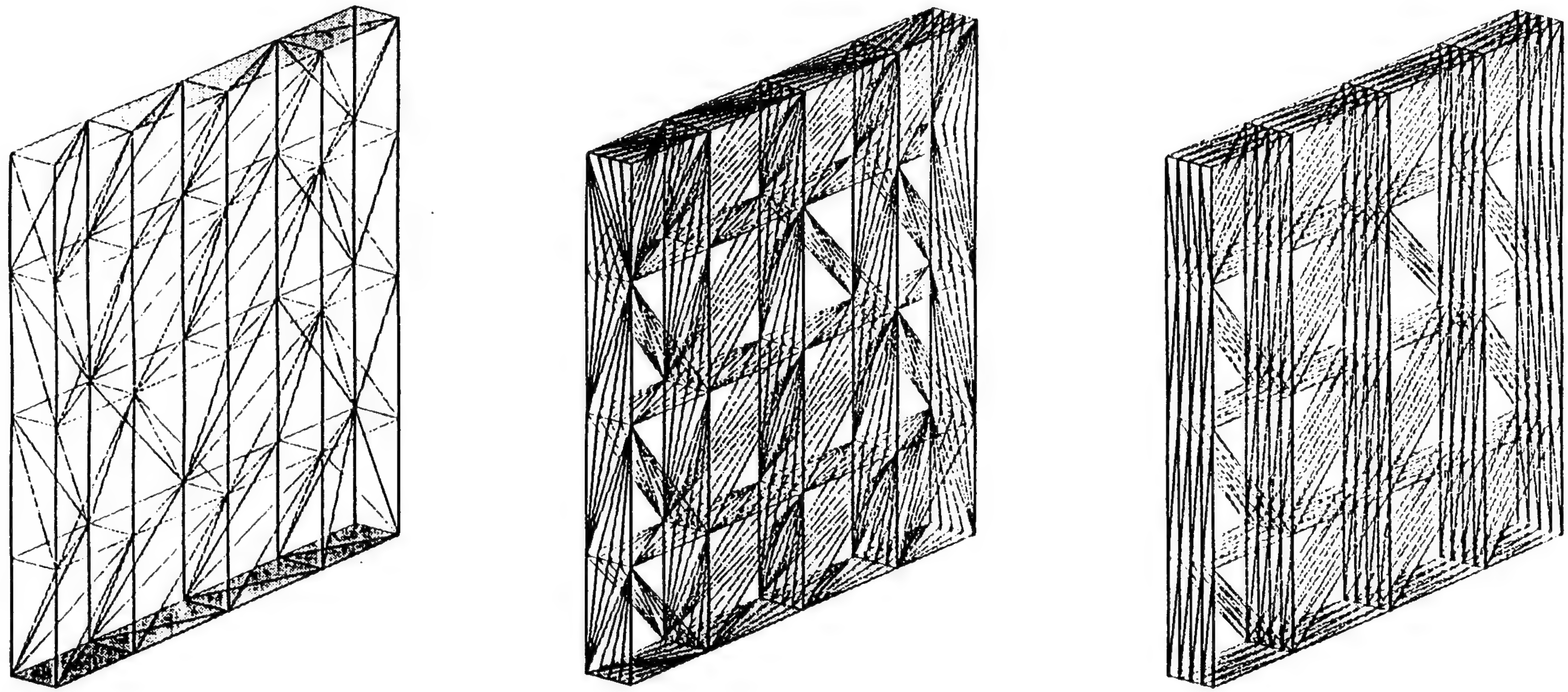


Transformation from "diagonal" to "zigzag" triangulation

SWAPPING EDGES (CONTD.)

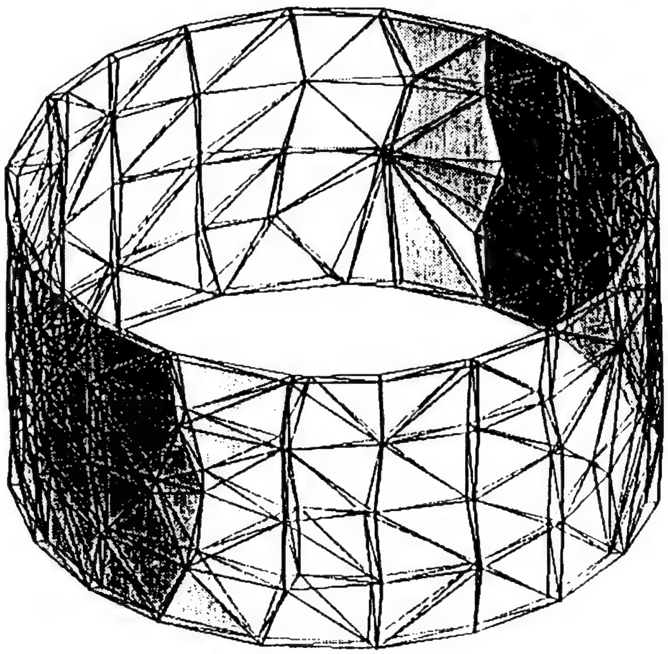
- Swapping sequence attempts to convert wedge with all "diagonal" faces to all "zigzag" faces
- Swapping done in templated sequence for efficiency whenever possible
- Must watch out for combinations of diagonal and zigzag triangulations that cannot be tetrahedronized
- If multiple edges in path have been split, recognize parts of mesh between opposite faces as wedges
- Pick swap with desired configuration and best large dihedral angle

EXAMPLE: *plate*

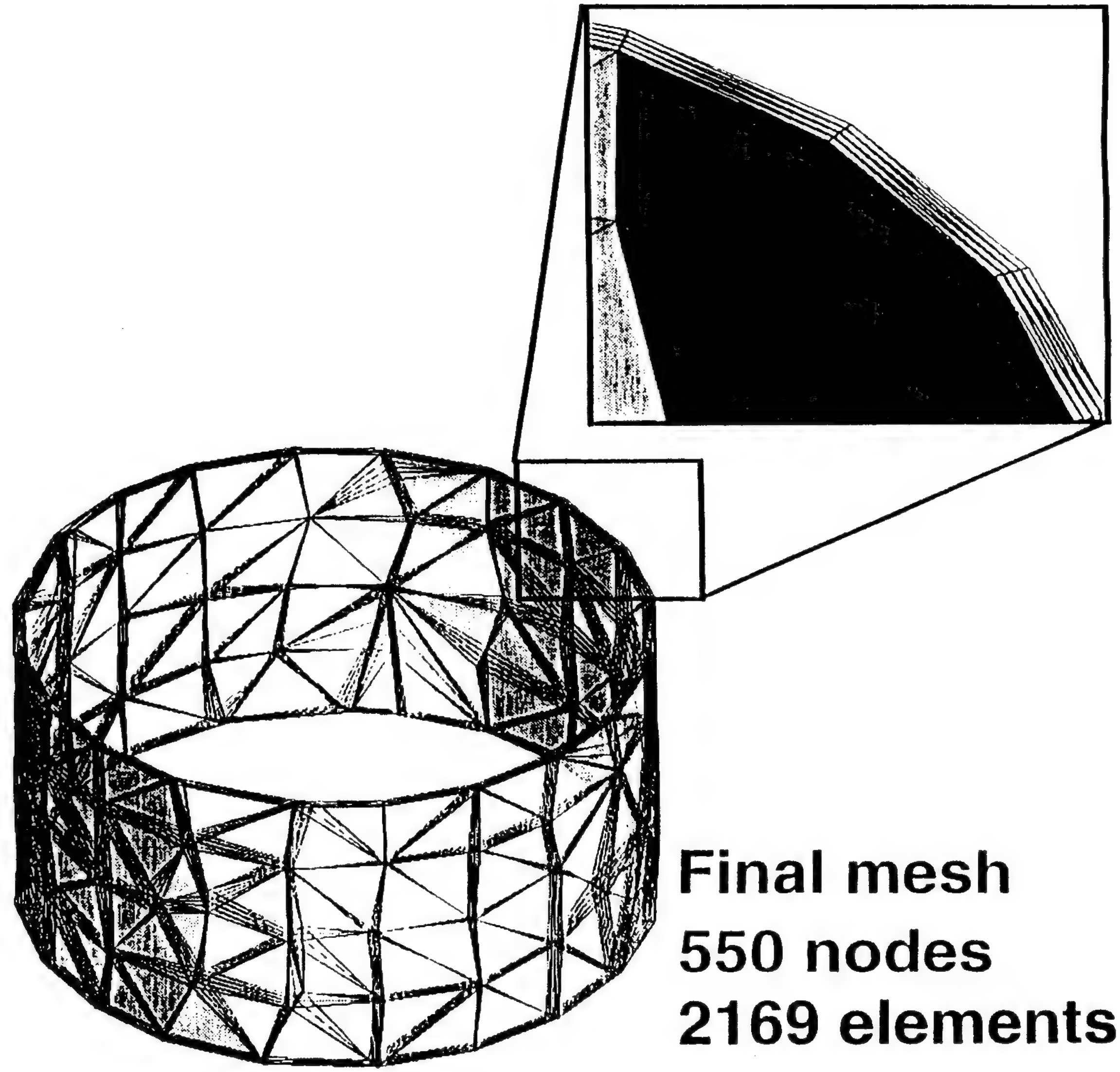


	Initial	"Split"	Final
Number of nodes	50	125	125
Number of elements	96	384	384
Largest Dihedral Angle	150	118	96

EXAMPLE: *Ring*

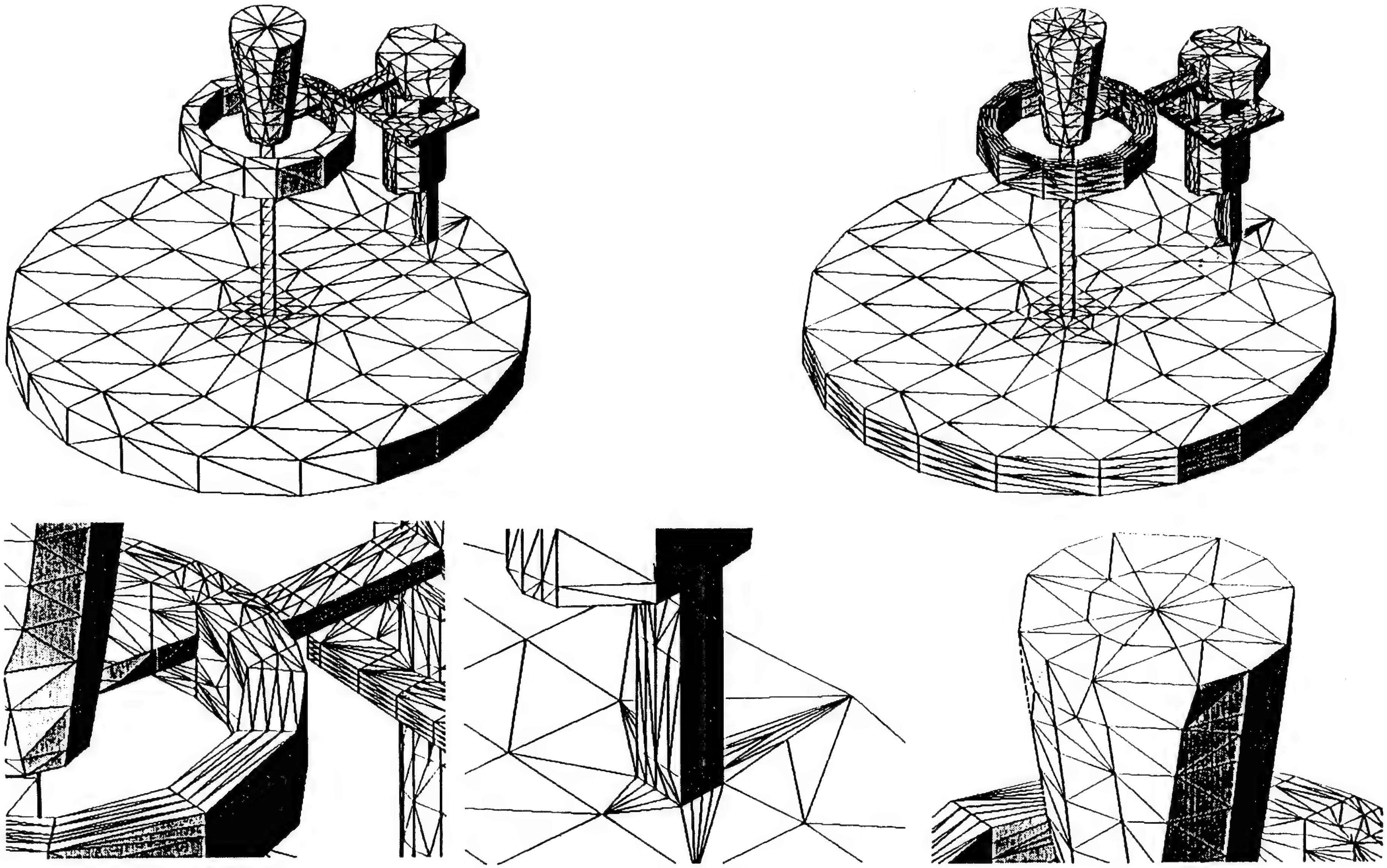


Initial mesh
217 nodes
520 elements

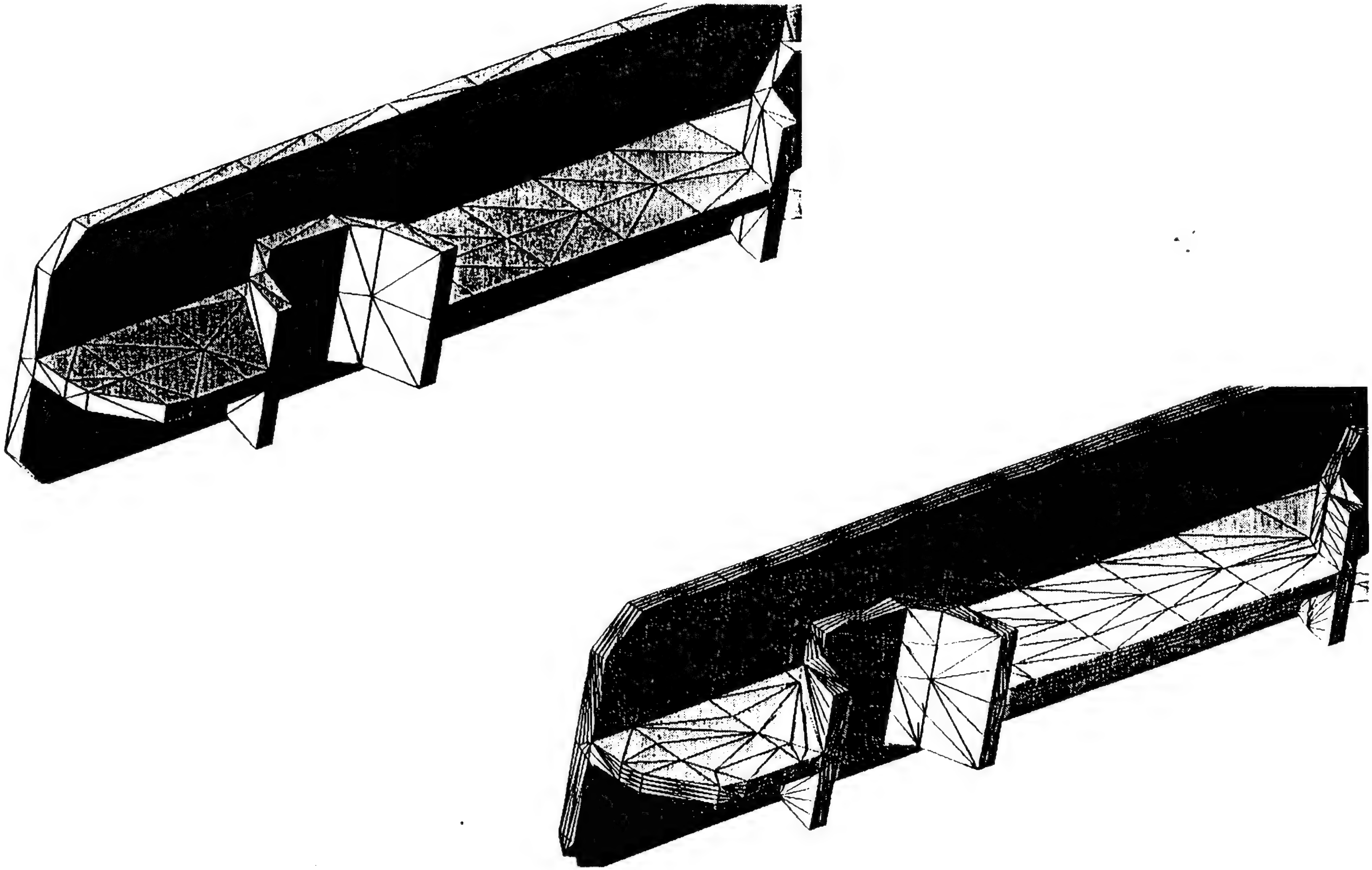


Final mesh
550 nodes
2169 elements

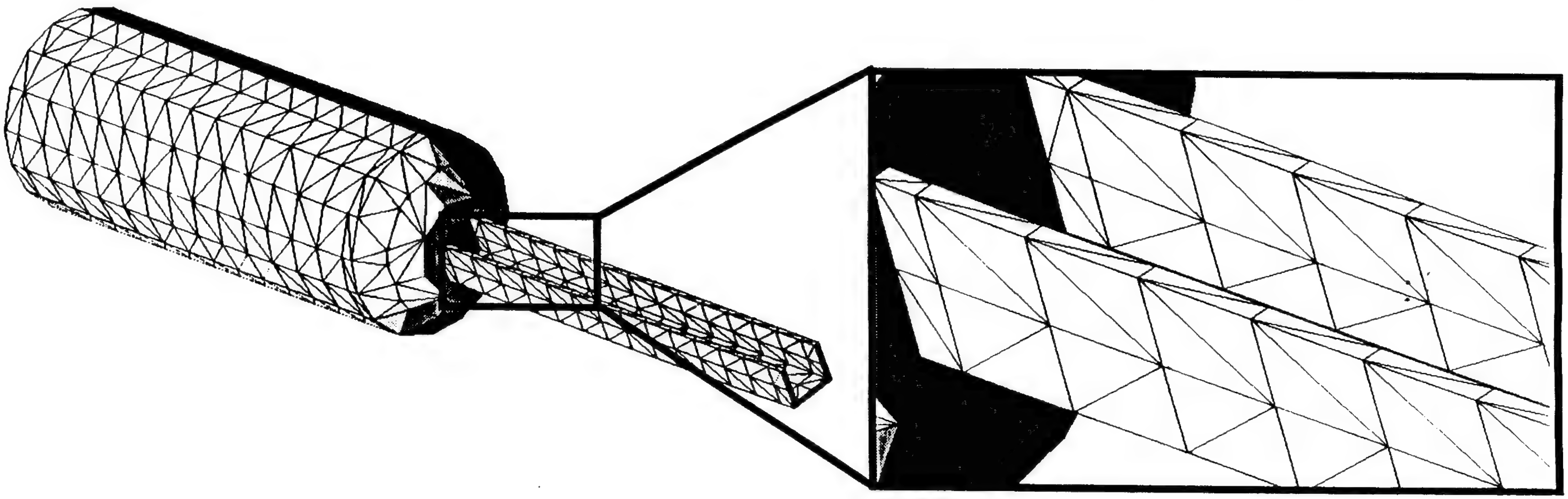
EXAMPLE: ICCA model *full_solid*

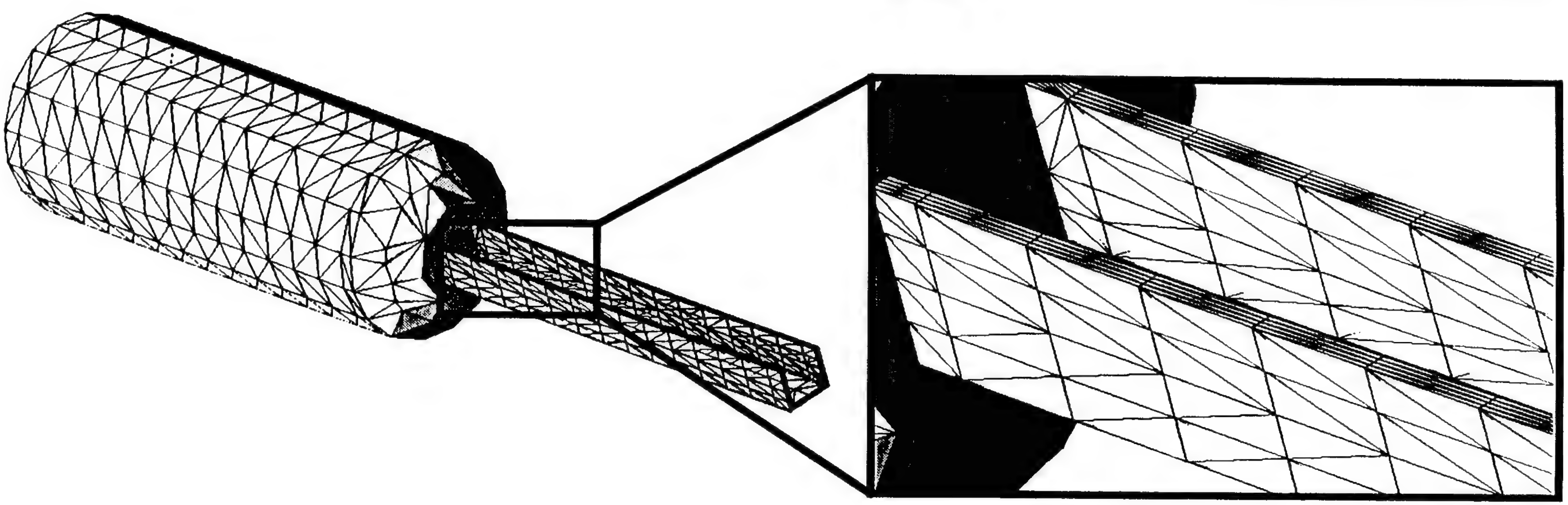


EXAMPLE: ICCA model *bracket*



EXAMPLE: CENTRIC DIE MODEL





Future work

- Capability to deal with multi-material models
- Ability to convert all diagonal triangulations to zigzag
- Better control over large dihedral angles
- More sophisticated mesh matching procedures to tailor starting mesh to specific requirements
- Ability to make wedges wherever possible and tetrahedronize to efficiently get final mesh
- Speed improvements

CLOSING REMARKS

Base tools to support automated adaptive analysis in deformation process design are known

First step in development of reliable procedures is examination of best combination of techniques

Some finite element modeling areas requiring more development include:

- Specialized mesh refinements in critical regions like shear bands
- 3–D geometry procedures when remeshes are required
- Effective transfer of solution variables when remeshes are required

Alternative Materials & Processes: Incomplete Design and Multi-process Optimization

by A. Chemaly
TechnoSoft, Inc., and
Visiting Research Scientist
Wright Laboratory

Abstract

Design research has long focused on geometric aspects of coupling product representation and analysis. More recently, with the advent of feature-based design, research has enabled the further coupling of processing knowledge. But our preoccupation with geometric issues has deferred the response to this need for coupling of knowledge about shape, material and process. We continue to be constrained by the less technical issues such as a standard set of features and a rigorous explication which is generic and unambiguous across applications and vendors. As a consequence, progress in automating process planning and material selection will continue to be beset with tedious and costly human intervention.

Materials and process design begins with part specifications and geometry, and requires the optimization (in terms of cost and time) of a sequence that describes setups, fixtures, and processing details, e.g., in the machining process step: operations, tooling, machine data, material properties, together with step-by-step process visualization . . . beginning with the raw material and ending with the final shape. But the more challenging goal is to optimize across all processing steps, while comparing alternative shapes, materials, and/or processes. While the spirit of Intelligent Knowledge-based Engineering (KBE) is to facilitate multi-disciplined team participation, the depth and complexity of the design environment, multiple and competing materials and processes, has become an obstacle to widespread practice.

The current trend in engineering automation systems is the application of artificial intelligence techniques to automate the production of customized machined parts by integrating the product design/constraints together with material and processing knowledge. The intelligent KBE user interface must go beyond simple part geometry and tolerances to include functional and processing constraints along with material, if desired. The part geometry is typically represented in terms of generic features such as pocket, slot, boss, etc. along with user-defined or customized features called

profiles. All features are represented in terms of objects that can also have non-geometric properties to aid in the process plan generation. For example, the raw stock object would have, in addition to its geometric features, property attributes such as hardness, yield strength, impact strength, operating temperature.

Each feature (geometric and non-geometric) of the product design is defined as an 'intelligent' object, i.e., with all the rules required for its creation and interaction with other objects. The system enforces design rules such as over or under constrained designs and assists in the process plan generation by interrogating the processing knowledge base. Unlike the usual KBE (parametric CAD/CAM) system, these rules need not only be associated with the part's geometry, they could also relate to computational analysis results and/or processing resources. The part geometry is just another attribute of the intelligent KBE system, and instead of simply evaluating a proposed geometric design, an intelligent KBE system assists in the design creation process through three primary components: 1) objects/sub-objects, 2) knowledge bases, and 3) an evolving model.

A model is formed from objects which are the base primitives of a KBE system. An object often corresponds to a real world entity. For example, a crank-shaft is an object commonly used for mechanical design of engines. Part objects contain all the information necessary to model themselves. Objects vary in intelligence. Primitive ones merely know how to draw themselves. More intelligent or smart objects, use the knowledge bases to make specific design decisions regarding size, shape, position, manufacturing process, and so forth. Some objects even decide which other objects are applicable for this design. Objects can also be formed by grouping together other objects (sub-objects). For example, an engine object may be a grouping of a crank-shaft, a piston and cylinder assembly and other sub-objects representing various lower level component parts of an engine. The key point is that these objects are combined together to form and to describe a part model.

Knowledge bases capture a set of object associations and relations to represent the domain knowledge of a product. The KBE system provides organized access to this knowledge base to improve the design process. By automating the interaction of these knowledge bases, the design/evaluate/

revise design loops can be minimized or eliminated. The interaction is governed by the relations between objects. Each object includes, and thus is constrained by, a set of rules which are applicable to the object which encapsulates them. The KBE engine must establish the order of dependency among the objects which therefore organizes the interaction between objects and their respective rules. Note, this capability of the KBE engine to establish dependency is what distinguishes rule-based systems and KBE systems.

The conceptual part model describes the part engineering process by combining the appropriate part objects. It describes the relationships between major and minor components in the product. The conceptual part model is often represented as a tree structure and represents the organizational schematic for the product design and process planning cycle. The basic philosophy behind this concept is that a design is composed from features as basic entities instead of primitives. In addition a feature based design environment supplies easy referencing and parametric association for feature properties. It supports cross-referencing of properties within the different features so that you can specify relations and place the results into other properties.

The concurrent engineering philosophy is an obvious goal related to teamwork. Alternative design specifications are required when a manufacturing problem is encountered in a product that cannot be built or is simply too expensive. Obtaining timely and quick feedback from the different engineering disciplines involved in the design process is the key to avoiding rework and shortening the product design cycle. Geometric reasoning enables the system to analyze part features to automatically generate parameters for process planning simulation and automation. These parameters enable the integration of the part design activity with manufacturing and inspection planning.

The part geometry, design specifications/properties, and the manufacturing/inspection plans are maintained and stored in a single object-oriented part model. This unique object-oriented part model enables engineers from various areas of expertise - design, analysis, manufacturing, inspection planning - to interact with the part representation simultaneously. As a consequence, a design team can capitalize on the near instantaneous 'what-if' benefits of concurrent engineering.

Implementation of a design process that efficiently incorporates the concepts of concurrent engineering is the real challenge.

KBE systems have demand driven calculation kernels. Demand driven calculation based systems maintain the associativity between the different objects and their properties relations. Unlike procedure oriented languages the user does not specify the order of computation, objects are instantiated and values are computed only when they are demanded. Demand-driven calculation is very efficient for selective analysis of large models to inquire about alternative results when using "what-if" changes to ascertain different property values. Properties and subobjects are not instantiated until they are demanded.

In object oriented modeling systems the part model reflects the hierarchy of the design intent. The part model structure incorporates the relation among all the various parts including the dependency and relation/ constraints among the properties. Dependencies are automatically tracked and are used to trigger the constraints to enforce the relations. Constraints and dependencies can be added after the parts and subparts are created. Therein, dependency tracking assists in augmenting the rules and constraints after the model have already been instantiated. Dependency backtracking automatically verifies rule violations and notifies properties to recompute according to constraints. Modification in any property automatically updates the necessary changes in the entire model to maintain consistency.

Much work remains in feature-based representation architectures to associate geometric design with process and material knowledge. This presentation addresses an extension to features which is predicated on an architecture for intelligent KBE technology, referred to as INCOMPLETE DESIGN, to not only integrate product, process and materials, but to also organize and apply evolving engineering and manufacturing knowledge. The ultimate capability is to capture valued expertise and avoid relearning costly mistakes, while capitalizing on past successes and identifying future opportunities.

In situ Process Design & Discovery

May 24:

8:30 An Evolving Computational Model for In-situ Material Aging Assessment
Y. Pao, Wright Laboratory Visiting Scientist

9:15 Materials Design using Pyramidal Networks
A. Jackson, Wright Laboratory Visiting Scientist

10:00 - 10:15 Break

10:15 A Rapid Supervised Learning Neural Network for Function Interpolation and Approximation
P. Chen, Wright Laboratory Visiting Scientist

An Evolving Computational Model for In-Situ Material Aging Assessment

Yoh-Han Pao
Case Western Reserve University,
and
Visiting Research Scientist
Wright Laboratory

Abstract

With the advent of self-improving computational models, in-flight self-assessment of a composite air or space craft's performance limits may become a reality. Based upon the air or space craft's specific experiences (self-monitored wear-out history), real-time evolution of a system performance limits model will continually or upon request inform the pilot or maintenance personnel of current and future performance limits and structural life remaining, as well as prediction of impending failure. The development of sensors to accomplish this task has been an ongoing effort for several years as part of the Air Force 'Smart Structures' initiative Polymer, metal and ceramic composites lend themselves well to this technology, since the sensors can be embedded at manufacture.

Like humans, individual aircraft are a product of their life experiences - a unique set of missions characterized by pilot-specific decisions regarding speeds and maneuvers. Also akin to humans, with use and age, the strength of materials for various components diminish with time. Analogous to type A humans, military aircraft are continually driven to achieve at their limits of performance. To operate safely at their limits these systems require very accurate assessment of the strength of the aircraft and/or individual components or to be informed of impending failure at any point during its useful life.

The computational method must have the potential to, in real-time, inform the pilot or cockpit crew of any structural limits, either due to progressive wear-out or instantaneous damage (projectile impact, etc) and the magnitude of those limits. This is useful in making an in-flight advisory of, for example, whether the aircraft should be abandoned, returned immediately to base, complete the mission with restrictions, or complete the mission with no restrictions. Once the mission is complete, the method can assess whether the structure needs immediate repair, can be flown with restrictions (and

the magnitude of any restrictions) to depot for repair, or can be flown without restrictions. Beyond these safety evaluations, potential cost savings may be realized with this 'inspect for cause' approach, as opposed to the 'inspect at interval' method now currently used.

In principle,the structural integrity and functional capability of structures made of composite materials can be monitored with the use of distributed networks of embedded or imprinted sensors. The challenge is to have the ability to make sense out of the vast arrays of information items afforded by such arrays. This presentation addresses neural network computing techniques for data compression and 'data fusion' which facilitates such rapid meaningful interpretation of bodies of in-situ data. Analysis of the Donaldson and Springer approach indicates that it is an effective one, and data on aging or cumulative damage can be attained. However there can be complex interaction between aging mechanisms on both microscopic and macroscopic scales. This paper describes network computational methods for 'feature extraction' and for 'data fusion' which can be applied to the present task of building an evolving model of the state of a composite structure. Illustrations of such application to benchmark data and to actual sensor data are described.

AN EVOLVING COMPUTATIONAL MODEL FOR IN-SITU MATERIALS AGING ASSESSMENT

YOH-HAN PAO

Case Western Reserve University
and
Visiting Research Scientist
Wright Laboratory

PRESENTATION TOPICS

1. OBJECTIVE OF WORK
2. THE MATERIALS PERSPECTIVE

   Possible to Detect Signs of Cumulative Damage

   Interacting Mechanisms

3. THE STRUCTURES PERSPECTIVE

   Interacting Mechanisms

   Managing and Interpreting Data from Net of Distributed Sensors

4. THE COMPUTATIONAL MODEL

   The Internal Representation Associative Memory and Display

   Illustrative Examples:

   Learning Concepts in Reduced-Dimension Space

   Interpreting Sensor Data

   Discovering Associations

5. FUTURE WORK

   Implement Computational Model of Distributed Net of Sensors

   Validatewith Real Data

   Support Development and Implementation of Methodology

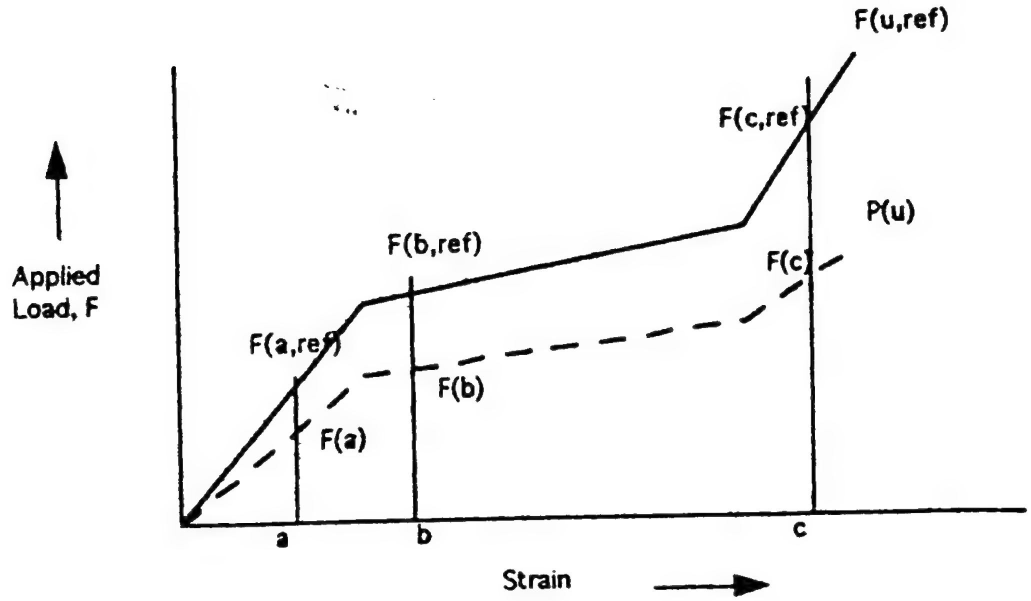


Figure 1. Illustrating The Idea of Monitoring Load Response By Comparing Ratio of Actual and Reference Loads, At Same Value of Strain

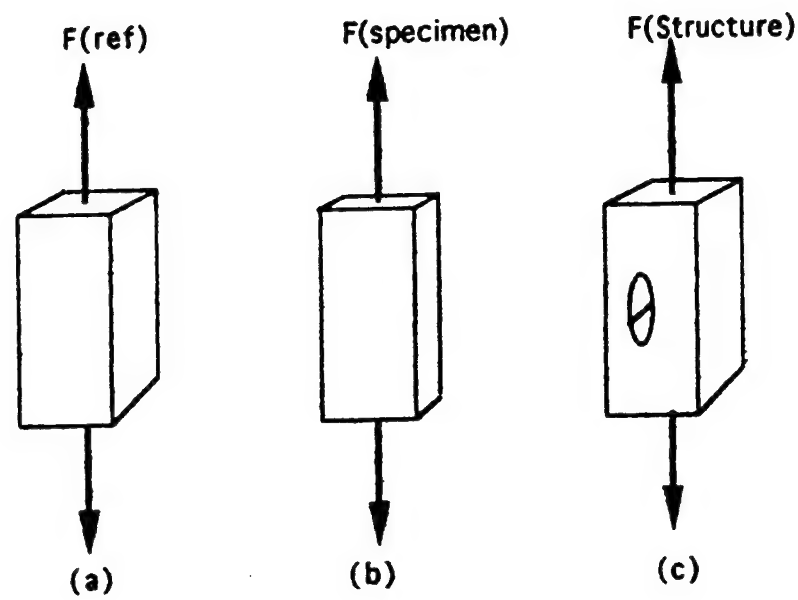


Figure 2 Illustrating the Difference Between Homogeneously Stressed Test Specimens and Non-Homogeneously Stressed Structures.

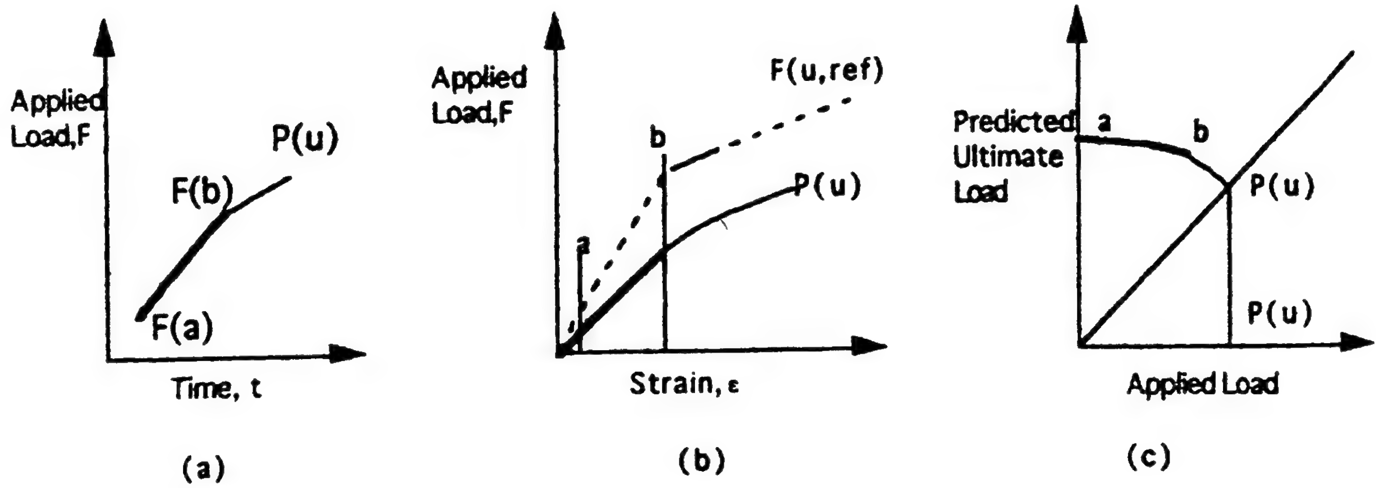
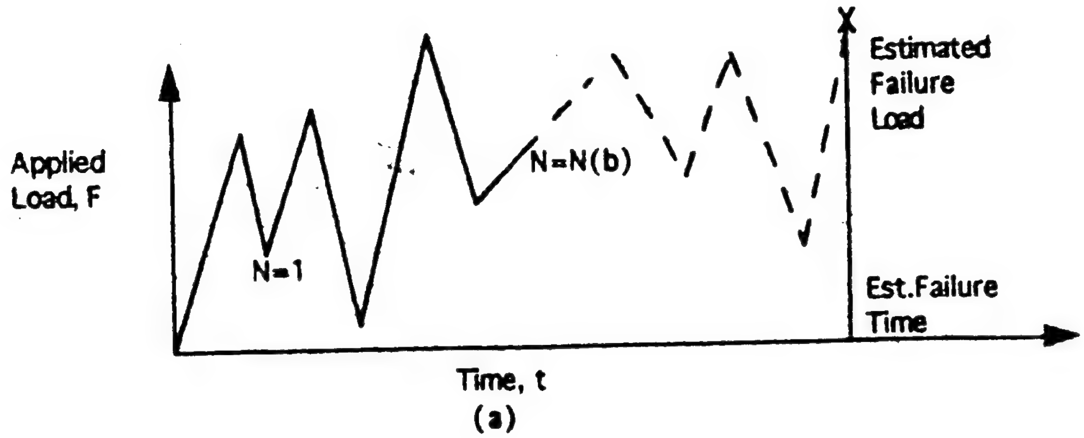


Figure 3 Illustrating The Idea of Comparing Observed Response to that of Reference and Using Extrapolated Ratio to Estimate Ultimate Failure Load for 'Structure'.
(Patterned after Figure 4 of Donaldson and Springer Reference).



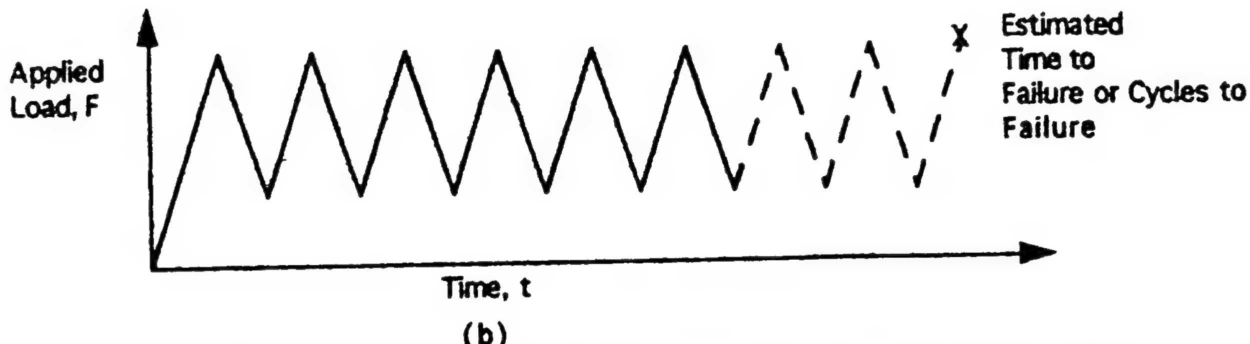


Figure 7 Predicting Fracture Failure in Random Cyclic Loading and In Constant Cyclic Loading

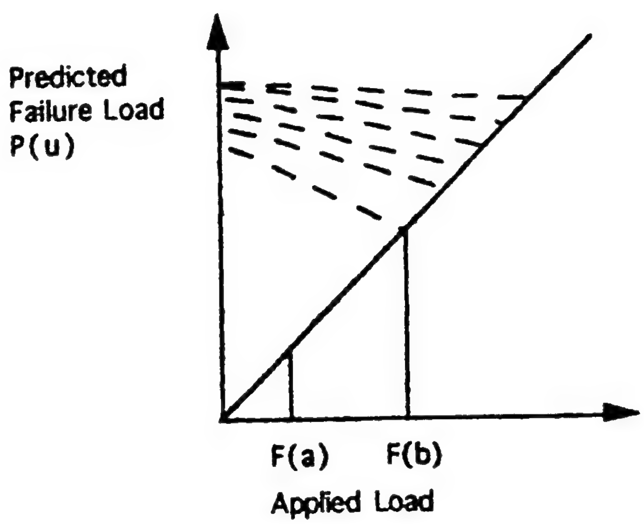


Figure 8 An Exxagerated Sketch of How The Predicted Failure Load Decreases With Every Cycle. But In Practice Extreme Resolution and Accuracy Would be Required.

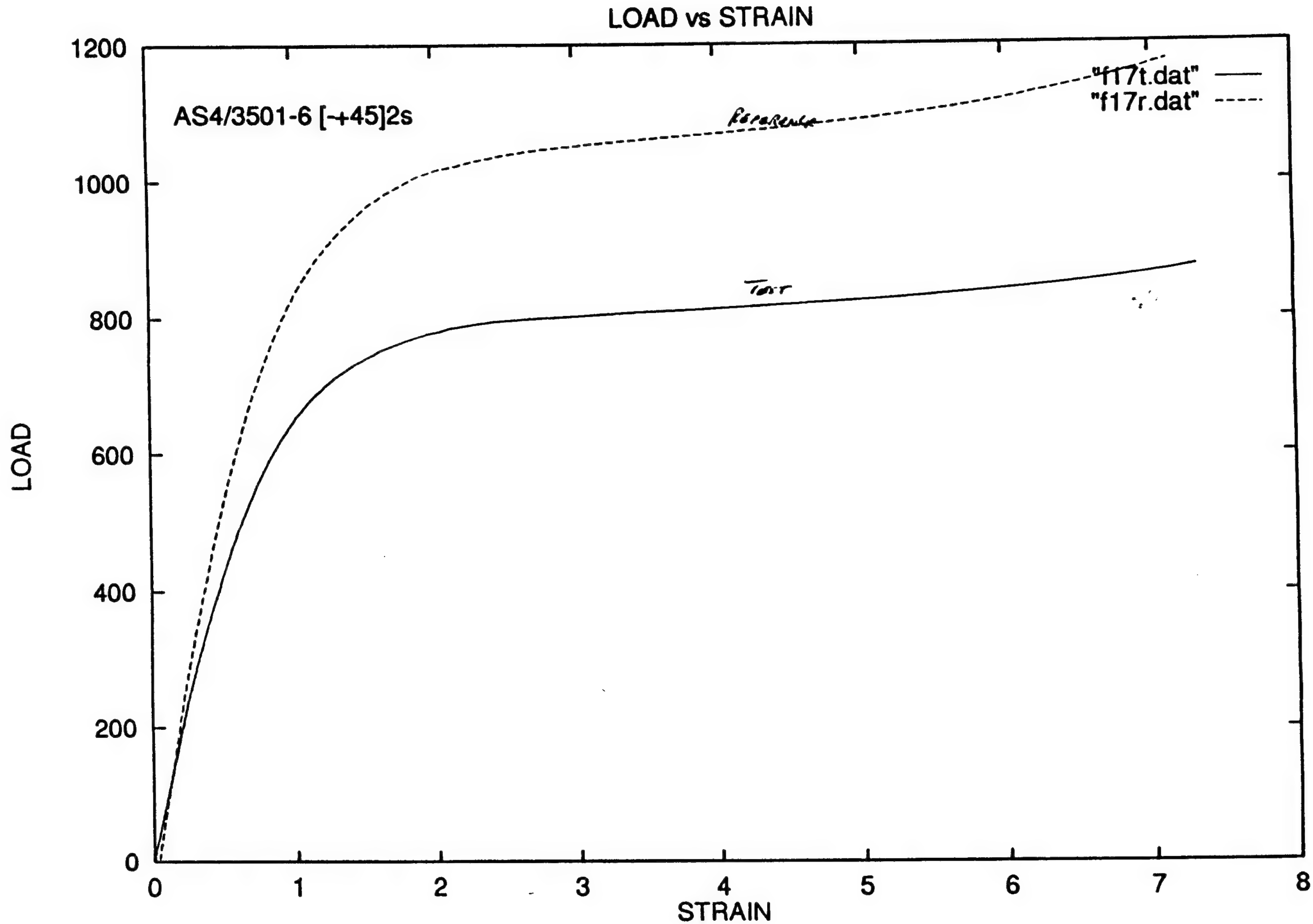


Figure 4 Illustration of Case Where Strains at Failure Are The Same

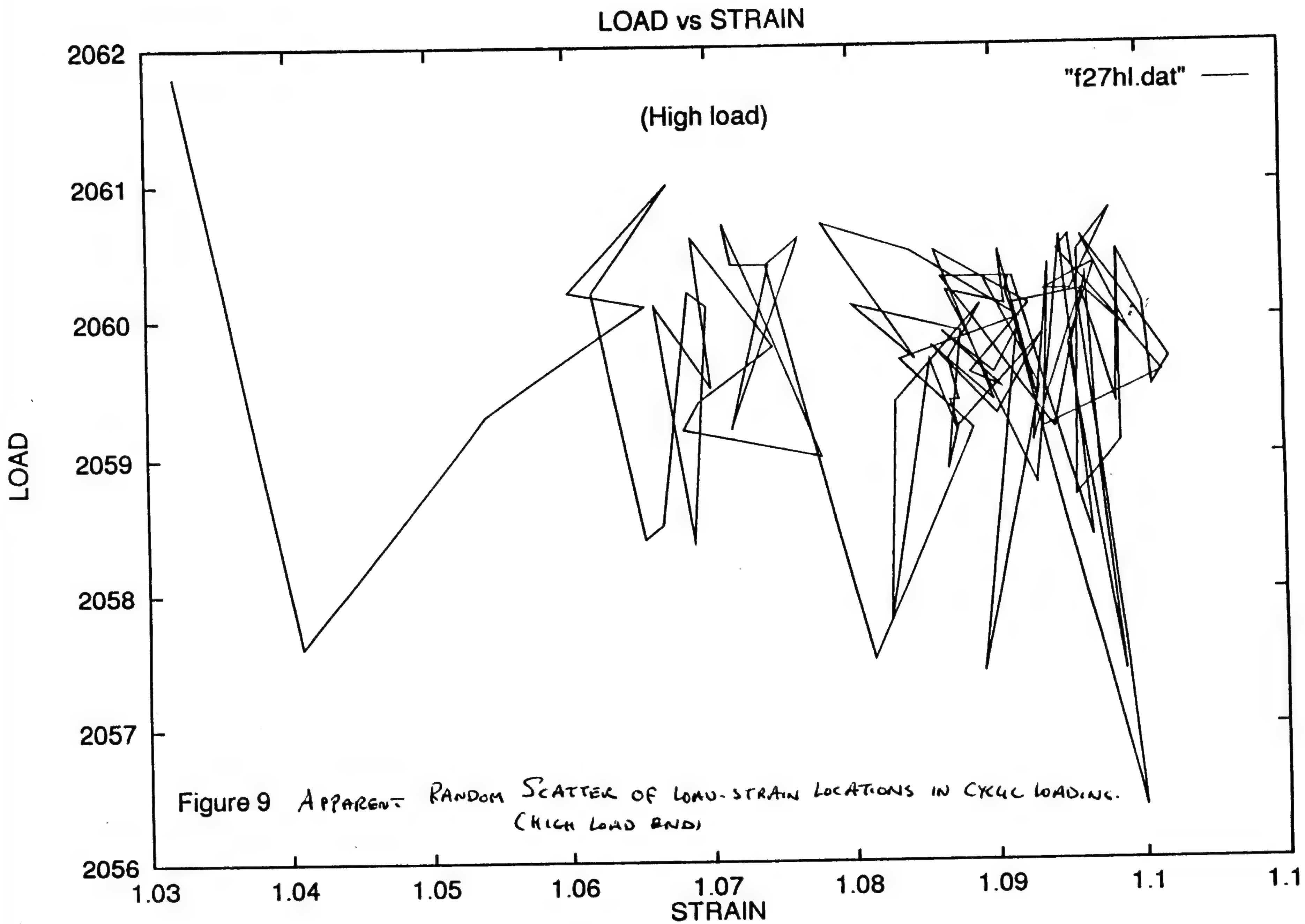


Figure 9 APPARENT RANDOM SCATTER OF LOAD-STRAIN LOCATIONS IN CYCLIC LOADING.
(HIGH LOAD END)

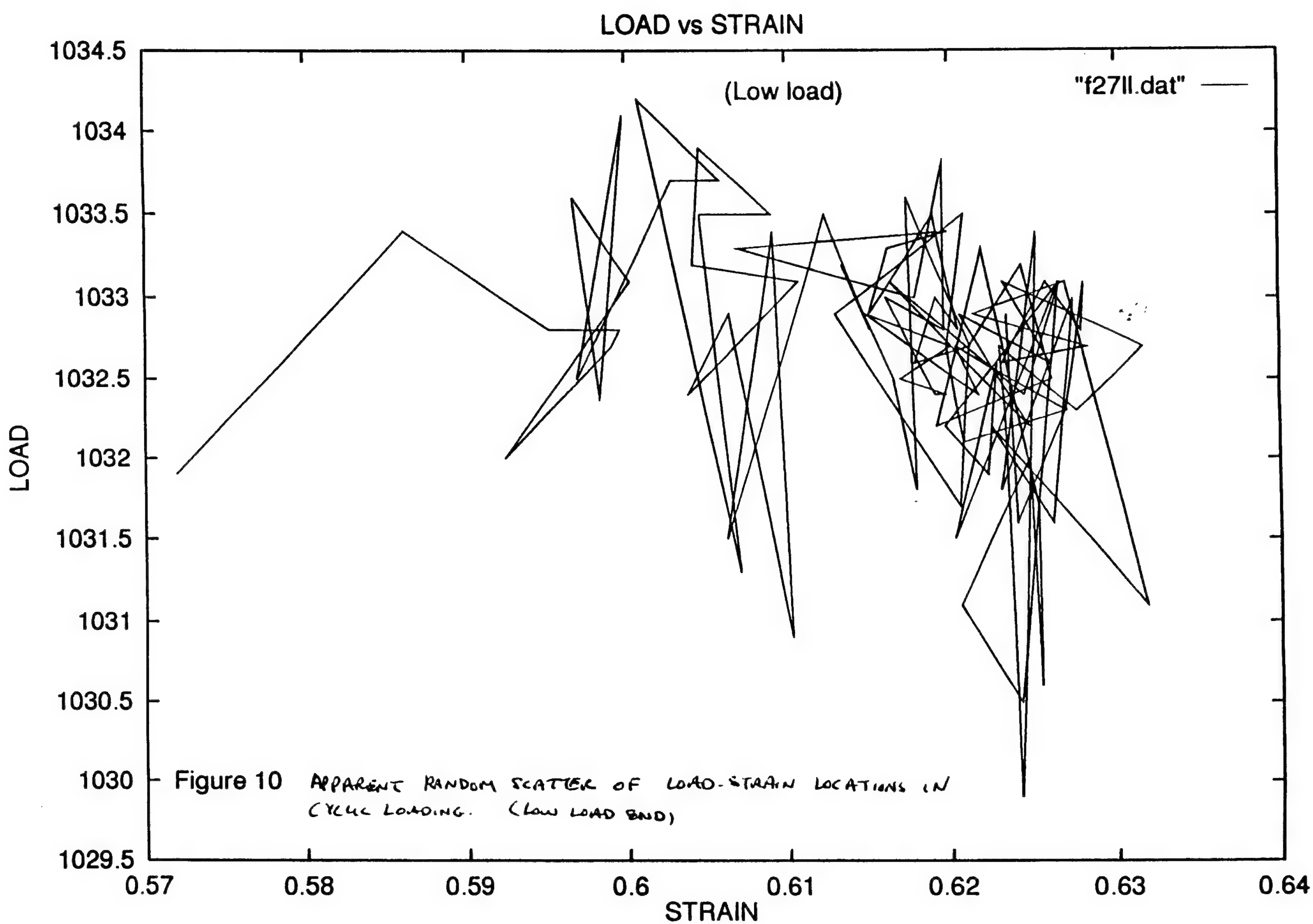


Figure 10 APPARENT RANDOM SCATTER OF LOAD-STRAIN LOCATIONS IN CYCLIC LOADING. (LOW LOAD BAND)

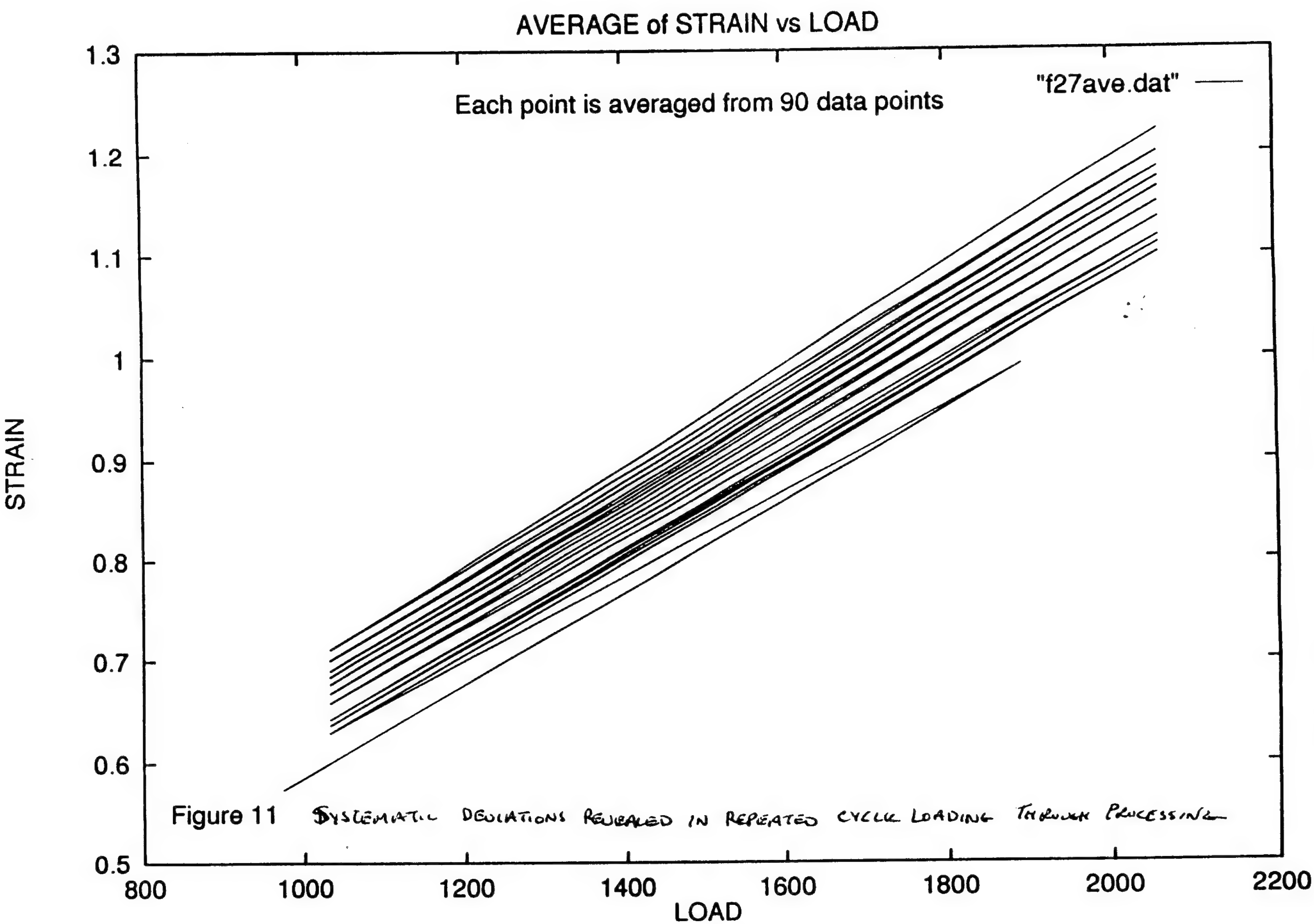
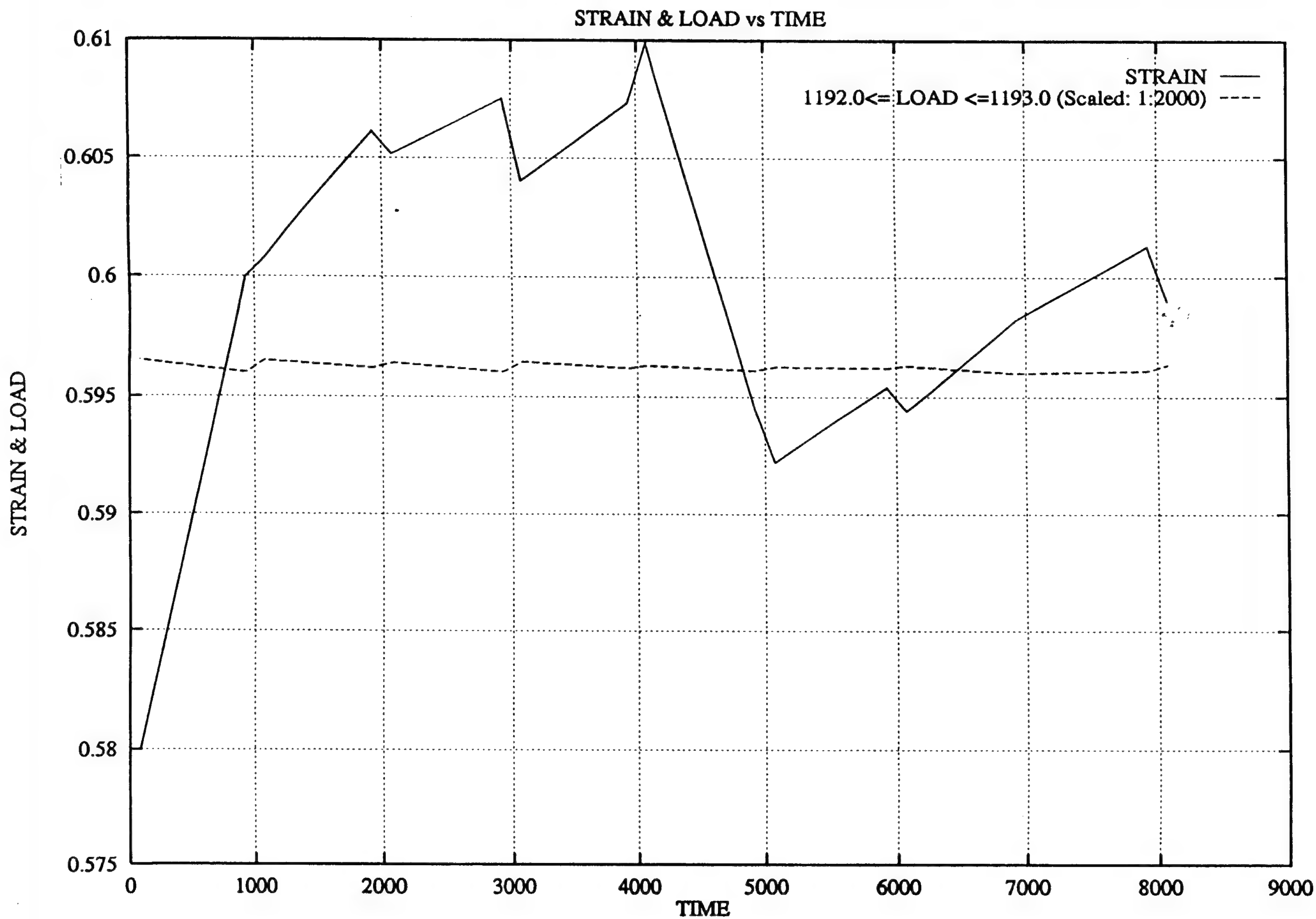
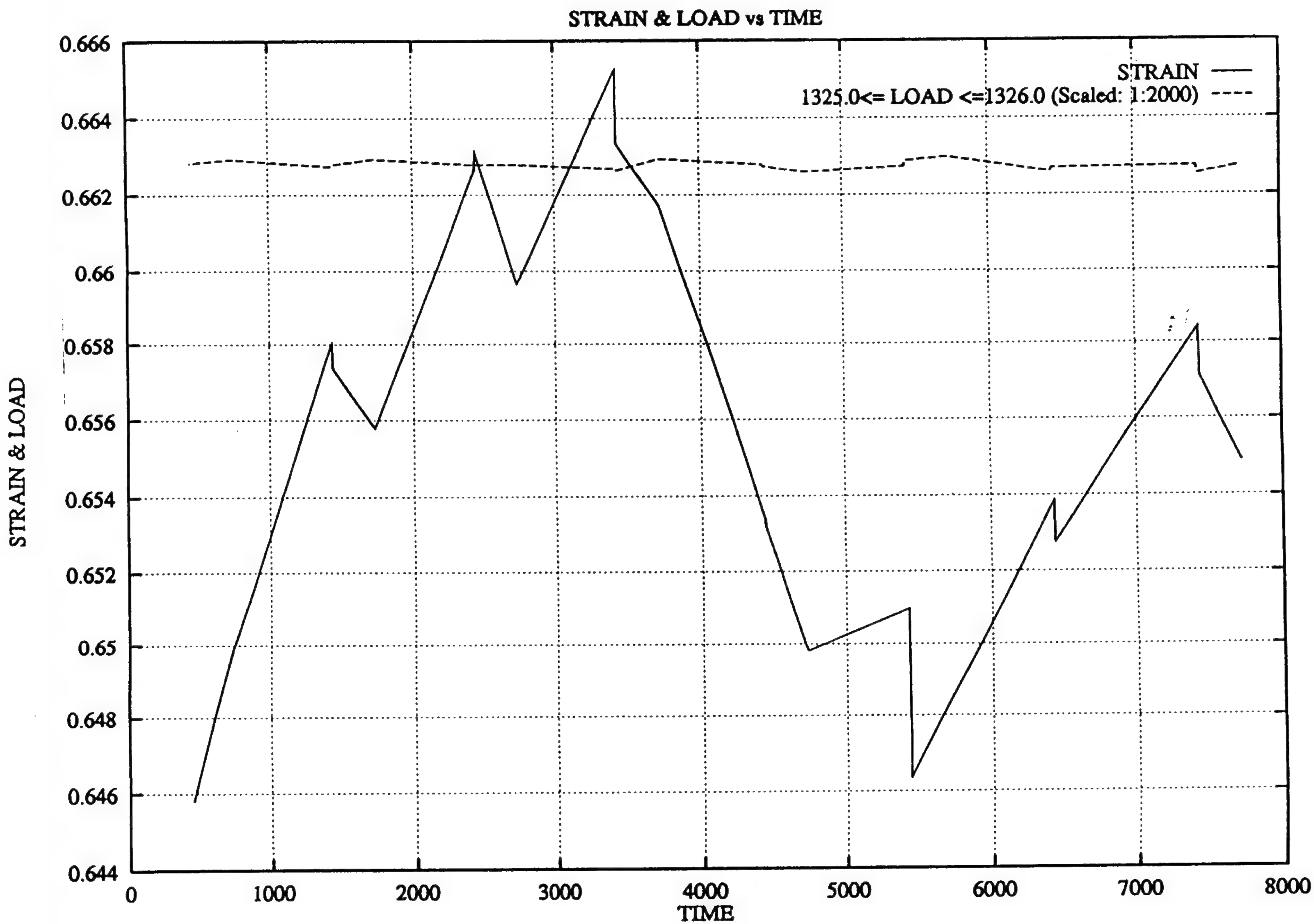


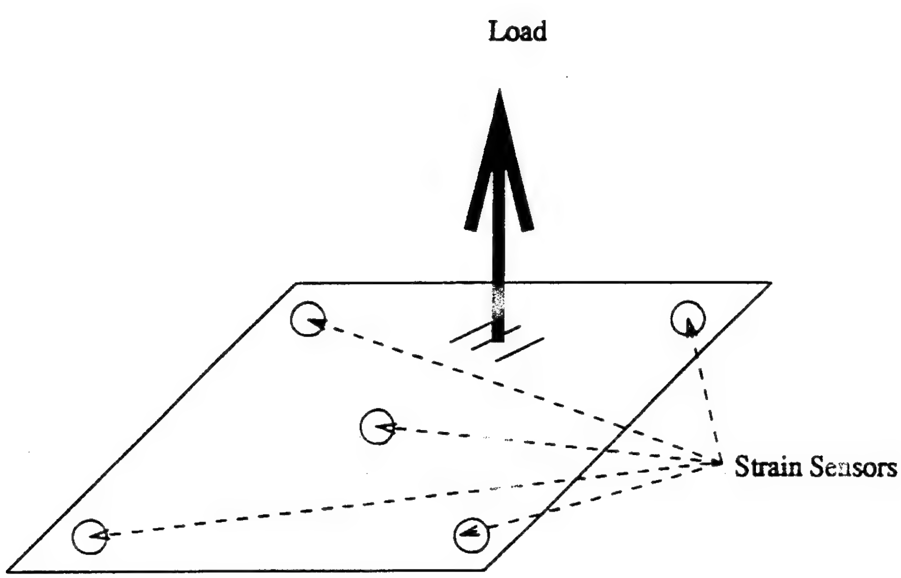
Figure 11 SYSTEMATIC DEVIATIONS REVEALED IN REPEATED CYCLIC LOADING THROUGH PROCESSING


STRAIN & LOAD vs TIME
STRAIN
1192.0<= LOAD <=1193.0 (Scaled: 1:2000)
STRAIN & LOAD
TIME
0.61
0.605
0.6
0.595
0.59
0.585
0.58
0.575
0
1000
2000
3000
4000
5000
6000
7000
8000
9000

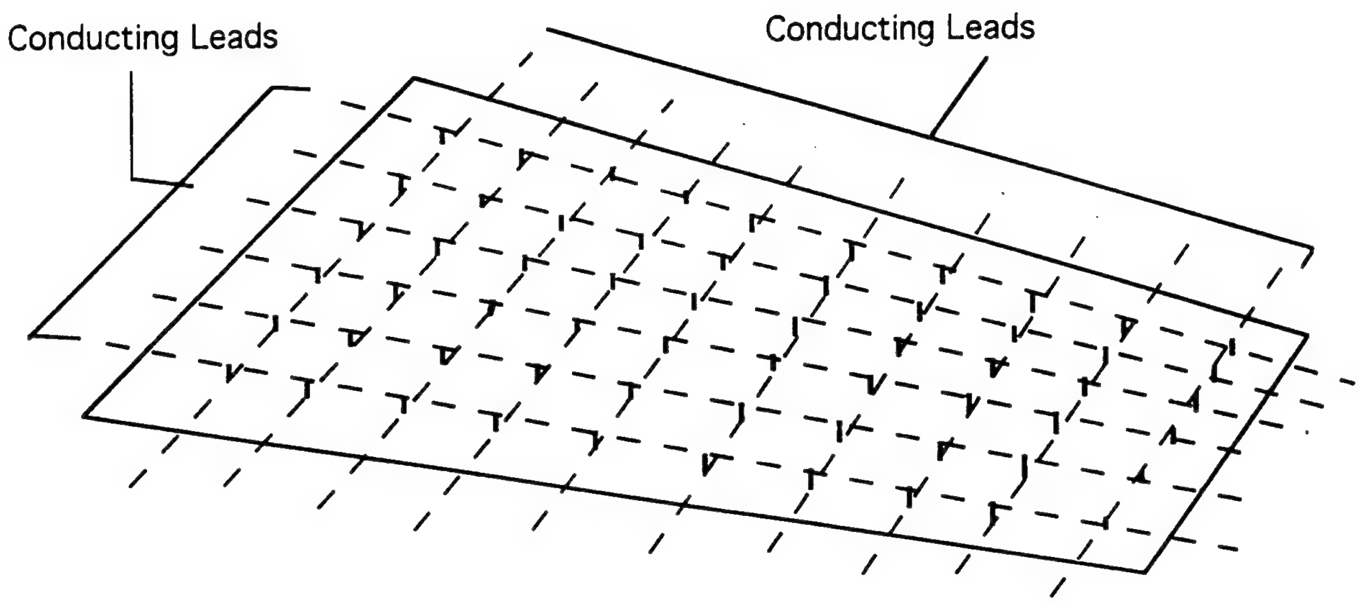
203


STRAIN & LOAD vs TIME
STRAIN
1325.0<= LOAD <=1326.0 (Scaled: 1:2000)
STRAIN & LOAD
TIME
0.666
0.664
0.662
0.66
0.658
0.656
0.654
0.652
0.65
0.648
0.646
0.644
0
1000
2000
3000
4000
5000
6000
7000
8000

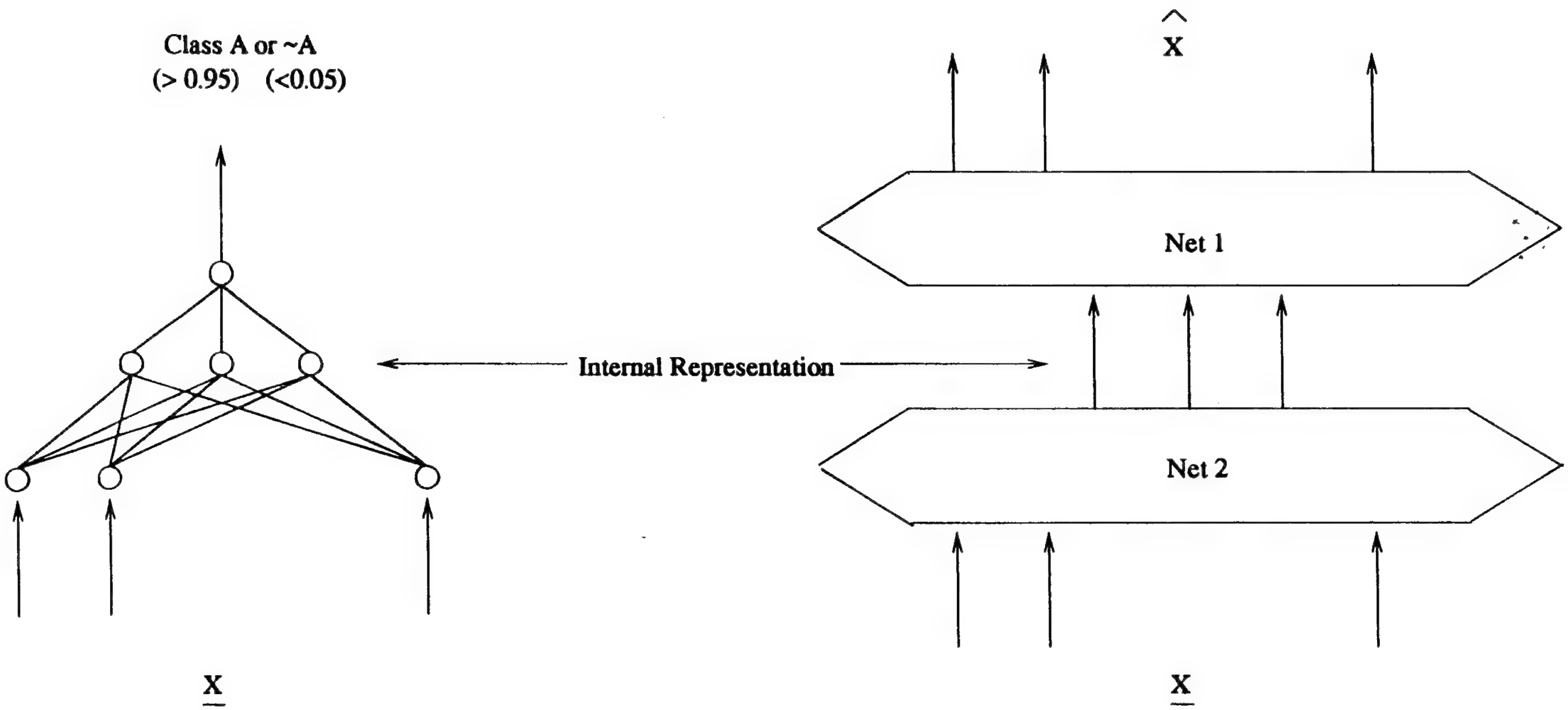
204



Processing Distributed Sensor Data



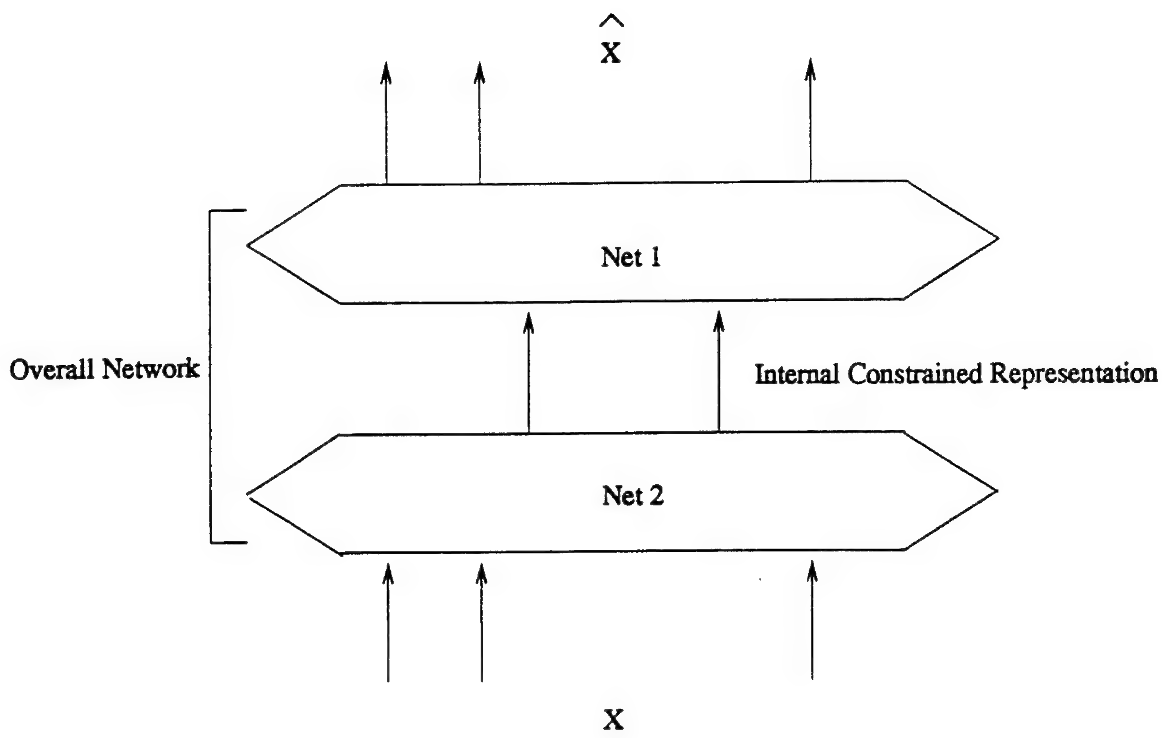
Schematic Illustration of Concept of Addressable Distributed Net
Of Embedded Strain Sensors



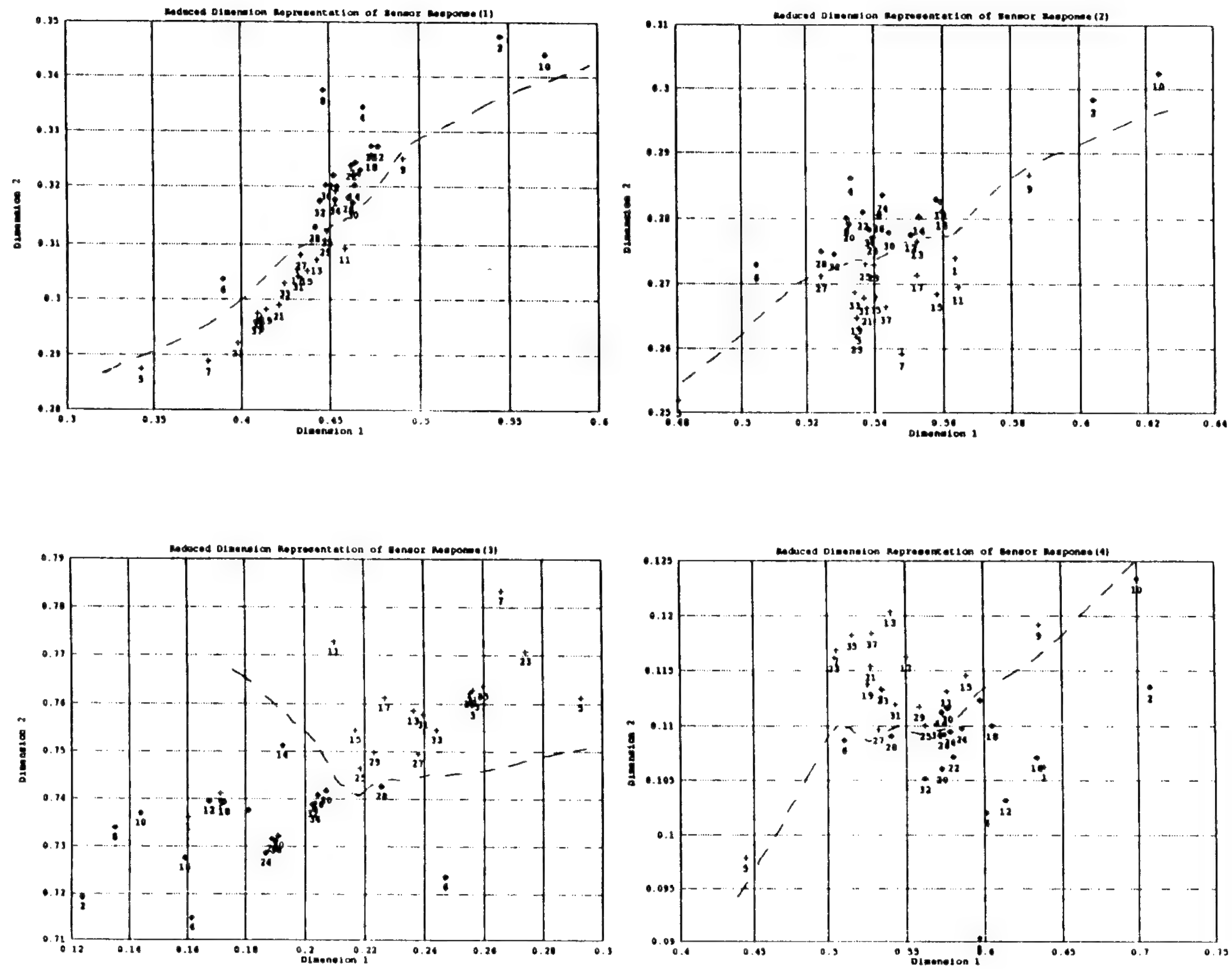
(a) Computing a 3-D Internal Representation

(b) Internal Representation Through Use of Auto-Association Net

Figure 1. Exploring The Hyporthesis That The Internal Representation Is Meaningful



Learning a Reduced Dimension Representation



Different Internal Representations Obtained with Different Initial States

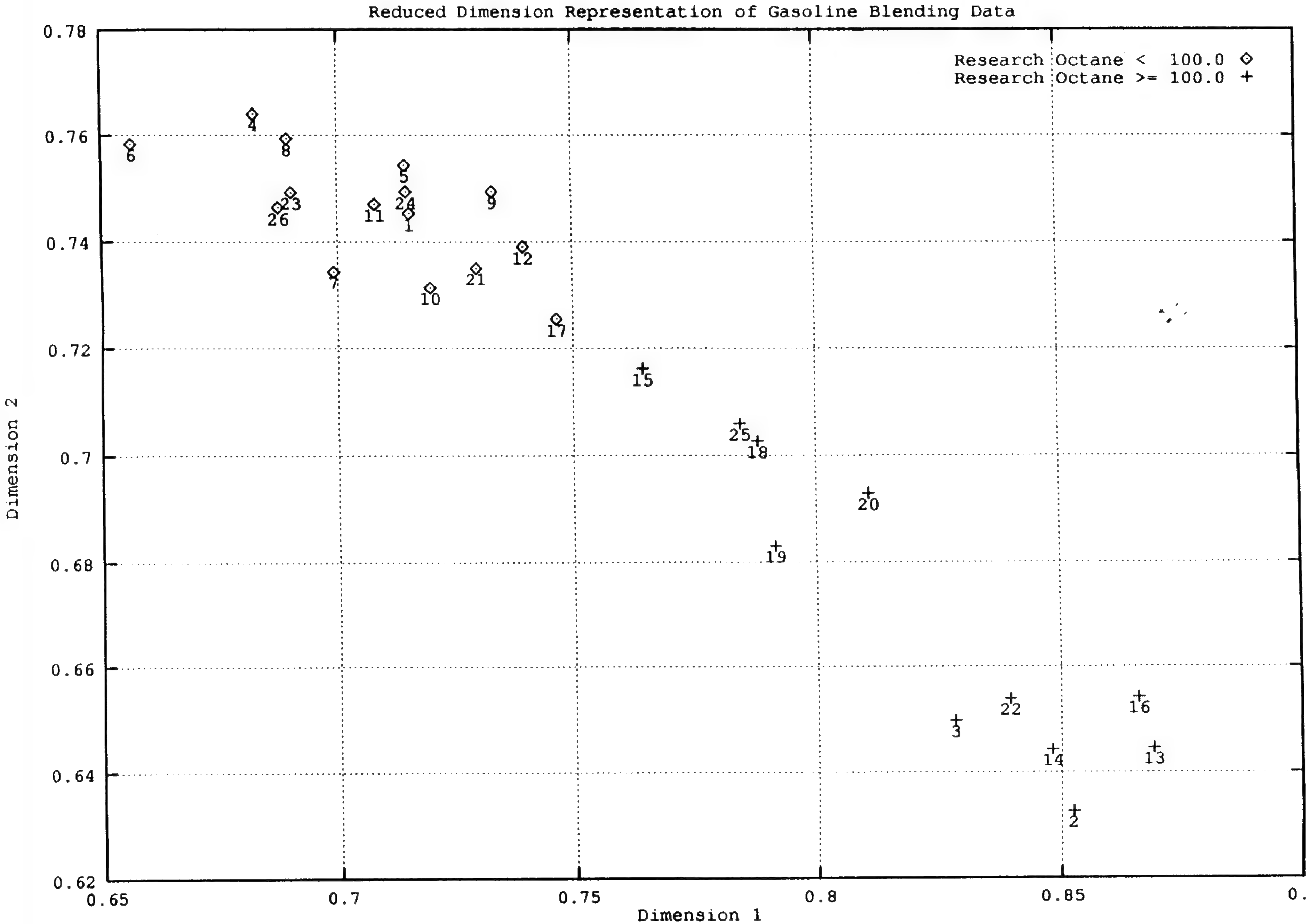
A Benchmark Body Of Data
Gasoline Blending Data

"Bayesian Control in Mixture Models"
By L. Mark Berliner
TECHNOMETRICS, NOVEMBER 1987,Vol. 29, No.4 P 455-460

No.	x1	x2	x3	x4	x5	y
1	0.000	0.000	0.350	0.600	0.600	100.0
2	0.000	0.300	0.100	0.000	0.600	101.0
3	0.000	0.300	0.000	0.100	0.600	100.0
4	0.150	0.150	0.100	0.600	0.000	97.3
5	0.150	0.000	0.150	0.600	0.100	97.8
6	0.000	0.300	0.049	0.600	0.051	96.7
7	0.000	0.300	0.000	0.489	0.211	97.0
8	0.150	0.127	0.023	0.600	0.100	97.3
9	0.150	0.000	0.311	0.539	0.000	99.7
10	0.000	0.300	0.285	0.415	0.000	99.8
11	0.000	0.080	0.350	0.570	0.000	100.0
12	0.150	0.150	0.266	0.434	0.000	99.5
13	0.150	0.150	0.082	0.018	0.600	101.9
14	0.000	0.158	0.142	0.100	0.600	100.7
15	0.000	0.000	0.300	0.416	0.239	100.9
16	0.150	0.034	0.116	0.100	0.600	101.2
17	0.068	0.121	0.175	0.444	0.192	98.2
18	0.067	0.098	0.234	0.332	0.270	100.5
19	0.000	0.300	0.192	0.208	0.300	100.6
20	0.150	0.150	0.174	0.226	0.300	100.6
21	0.075	0.225	0.276	0.424	0.000	99.1
22	0.075	0.225	0.000	0.100	0.600	100.4
23	0.000	0.126	0.174	0.600	0.100	98.4
24	0.075	0.000	0.225	0.600	0.100	98.2
25	0.150	0.150	0.000	0.324	0.376	99.4
26	0.000	0.300	0.192	0.508	0.000	98.6

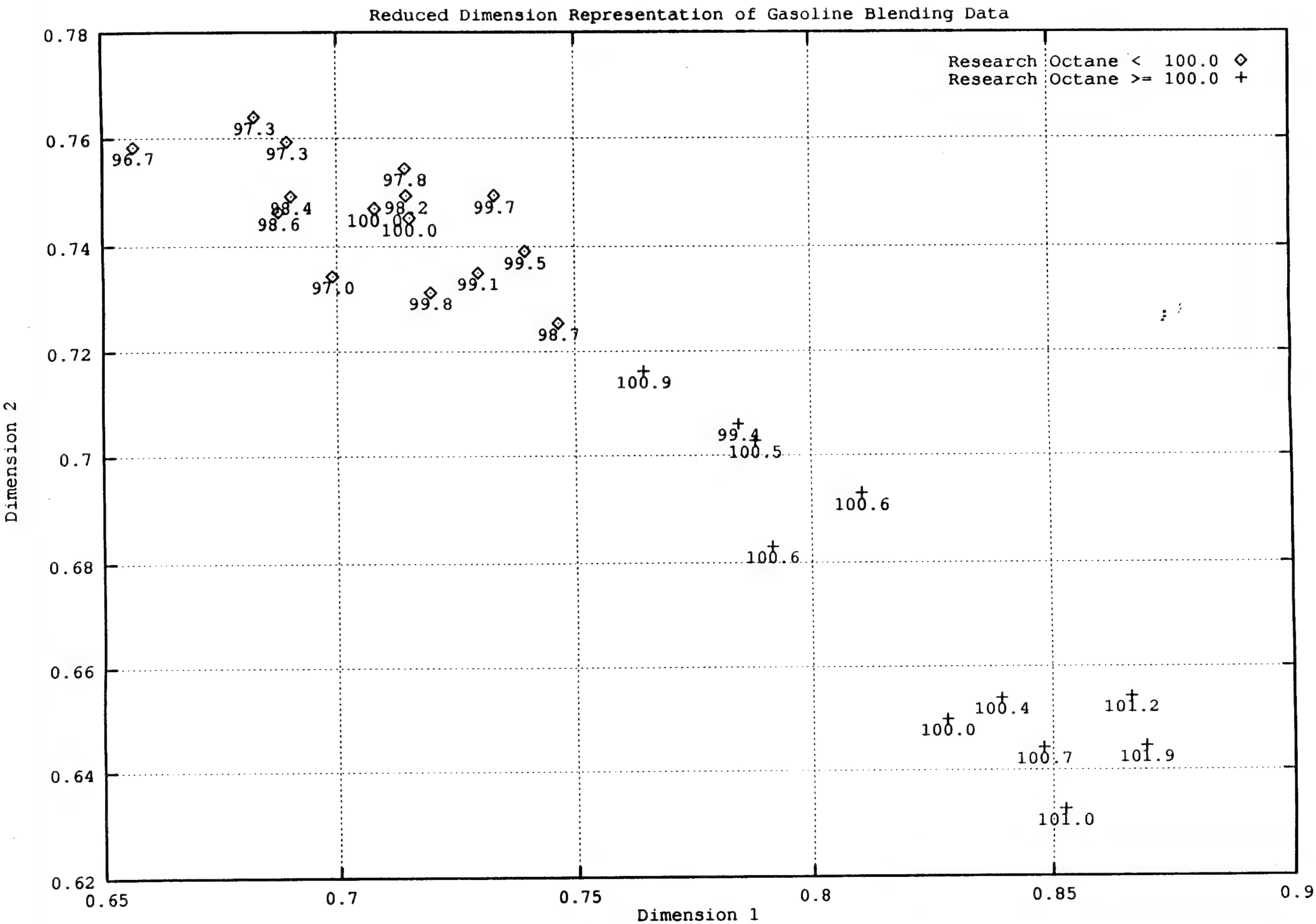
Table 1: Gasoline Blending Data

x1 = Butane
x2 = Isopenetane
x3 = Reformate
x4 = Cat Cracked
x5 = Alkylate
y = Research Cctane at 2.0 Grams of Lead/Gallon

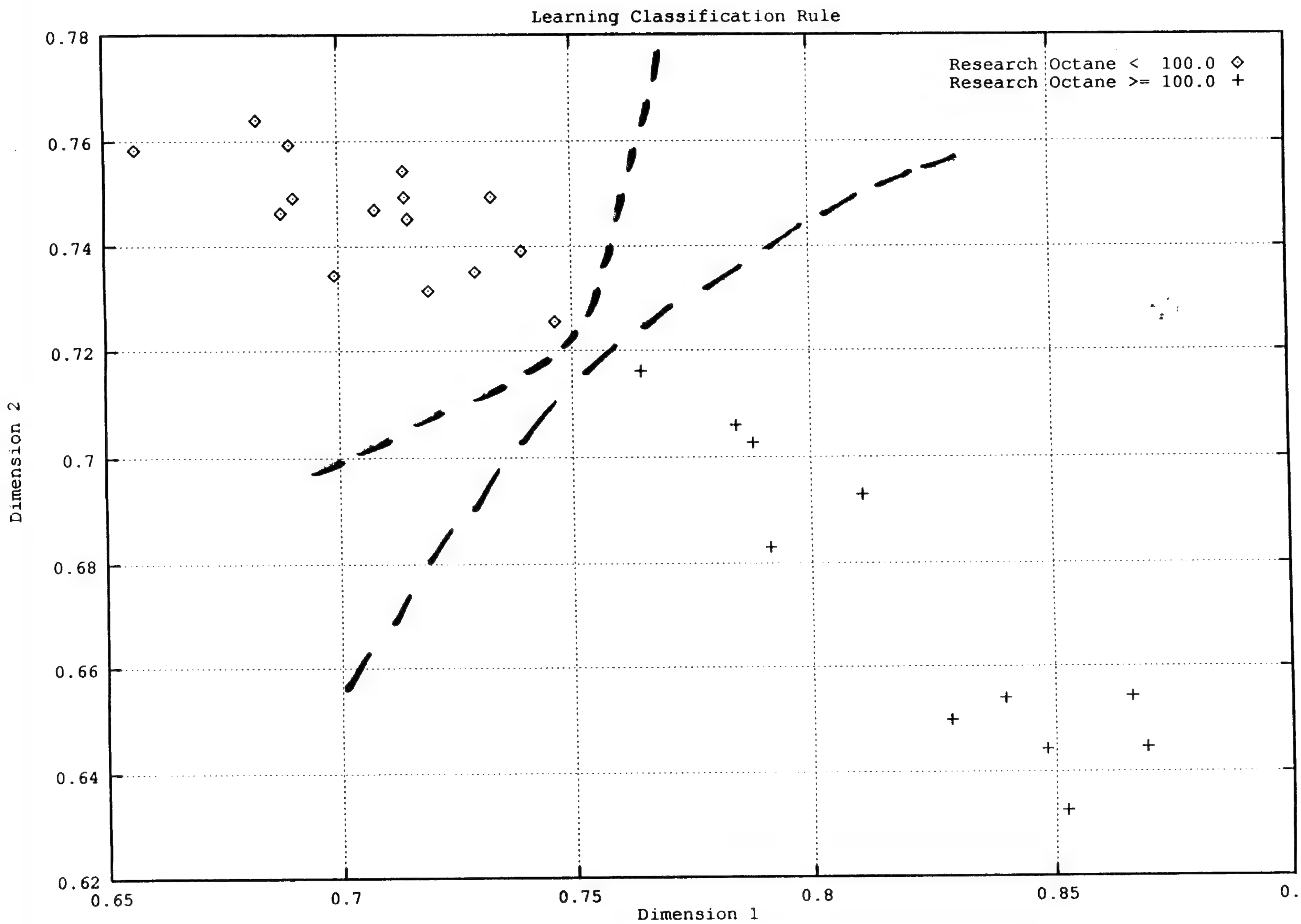


Reduced Dimension Representation of Gasoline Blending Data
Research Octane < 100.0
Research Octane >= 100.0
Dimension 1
Dimension 2
0.65
0.7
0.75
0.8
0.85
0.9
0.62
0.64
0.66
0.68
0.7
0.72
0.74
0.76
0.78
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

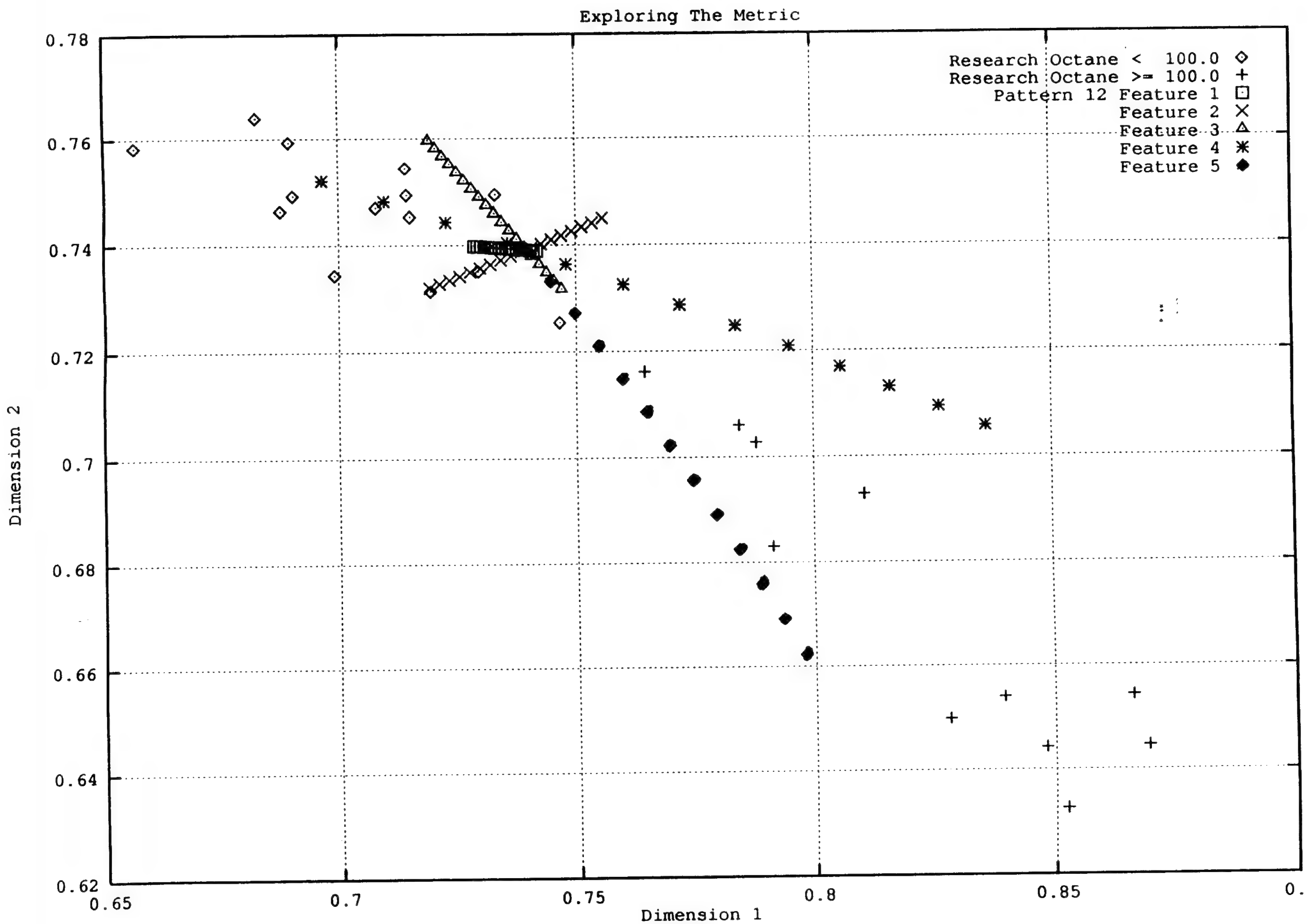
118

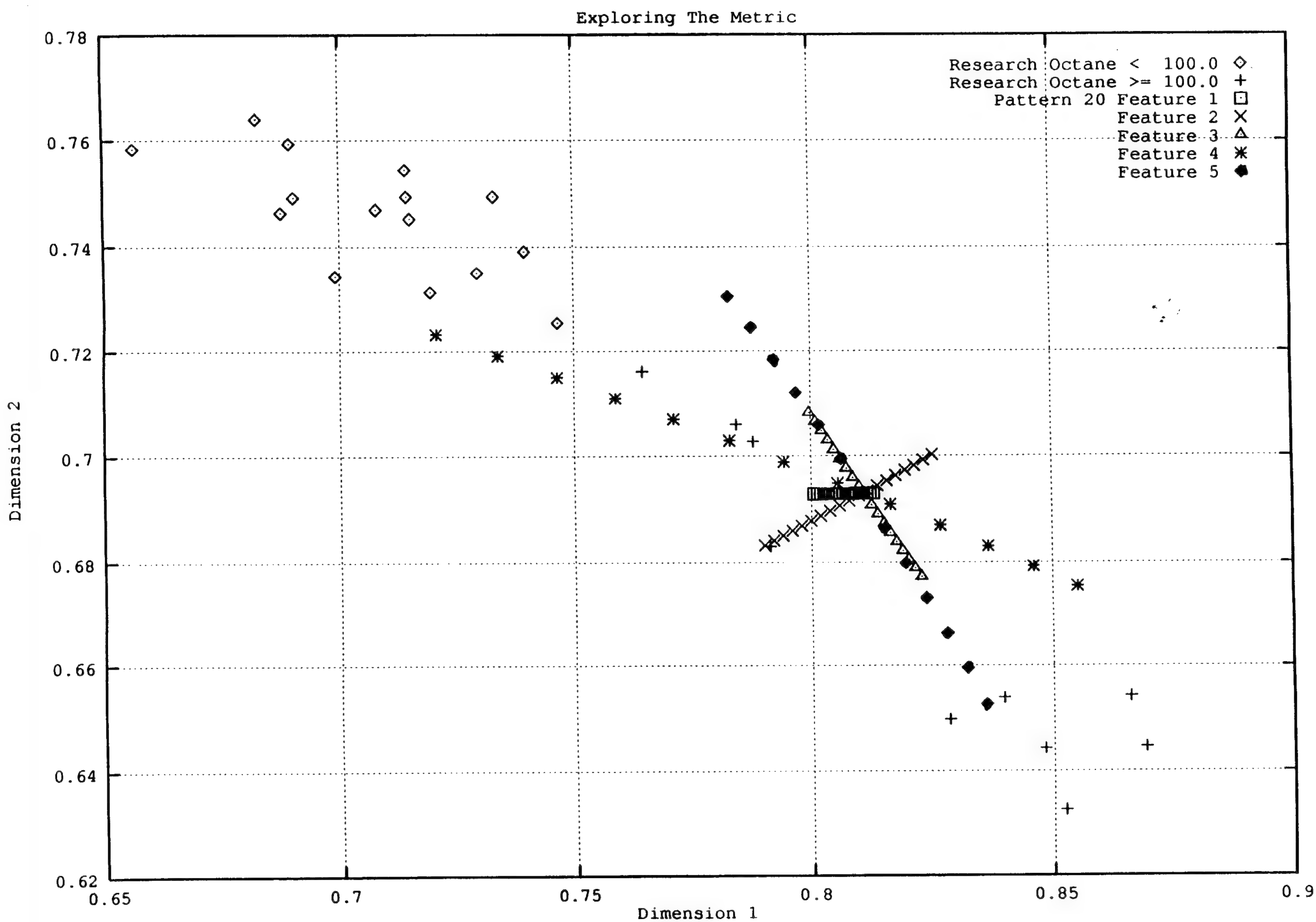

Reduced Dimension Representation of Gasoline Blending Data
Research Octane < 100.0
Research Octane >= 100.0
Dimension 1
Dimension 2
0.65
0.7
0.75
0.8
0.85
0.9
0.62
0.64
0.66
0.68
0.7
0.72
0.74
0.76
0.78
96.7
97.3
97.3
97.8
98.4
98.6
98.2
100.0
100.0
99.7
97.0
99.8
99.1
99.5
98.7
100.9
99.4
100.5
100.6
100.6
100.0
100.4
100.7
101.2
101.9
101.0

E16

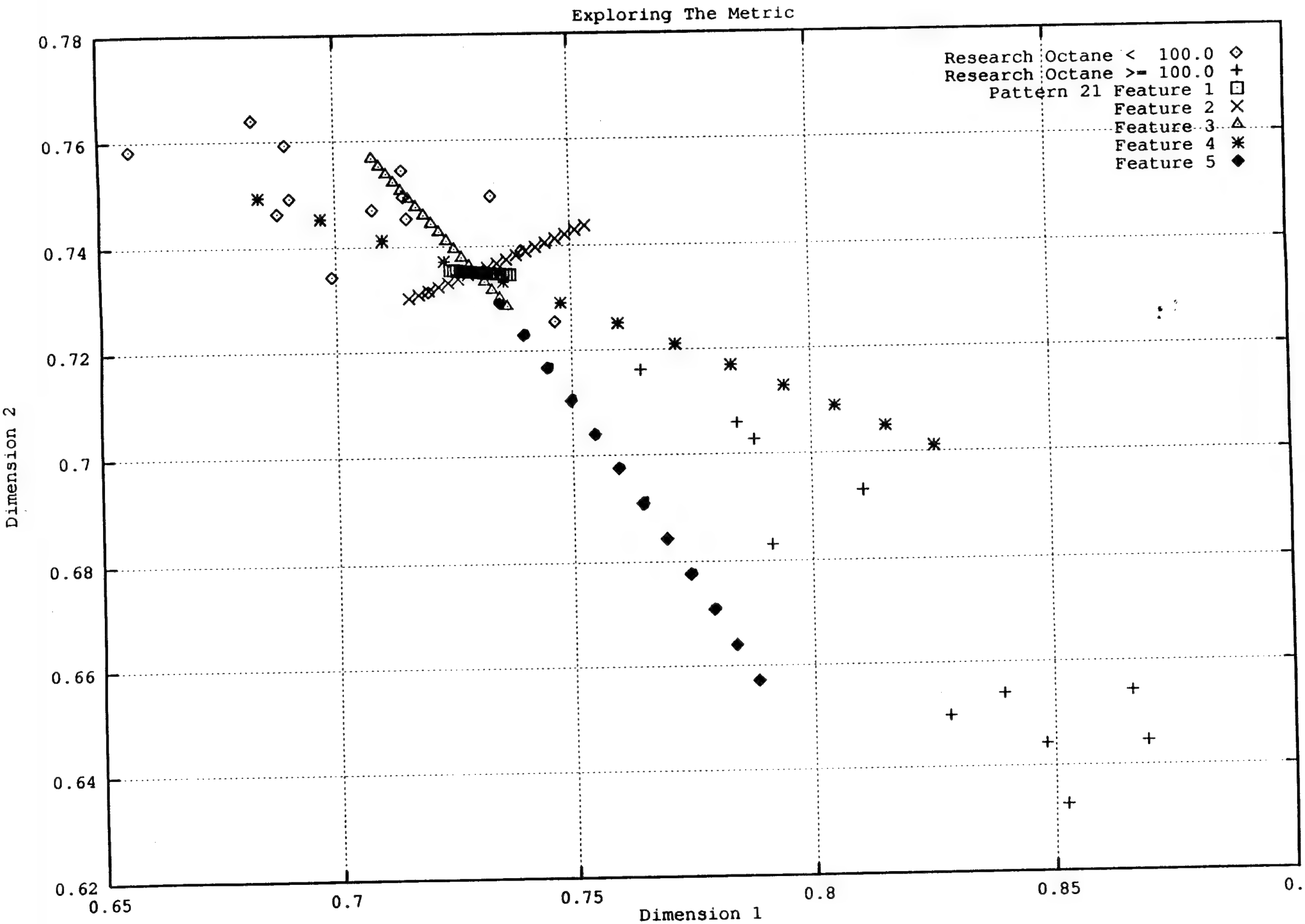

Learning Classification Rule
Research Octane < 100.0
Research Octane >= 100.0
Dimension 2
0.78
0.76
0.74
0.72
0.7
0.68
0.66
0.64
0.62
Dimension 1
0.65
0.7
0.75
0.8
0.85
0.9

213


Exploring The Metric
Research Octane < 100.0
Research Octane >= 100.0
Pattern 12 Feature 1
Feature 2
Feature 3
Feature 4
Feature 5
0.78
0.76
0.74
0.72
0.7
0.68
0.66
0.64
0.62
0.65
0.7
0.75
0.8
0.85
0.9
Dimension 1
Dimension 2

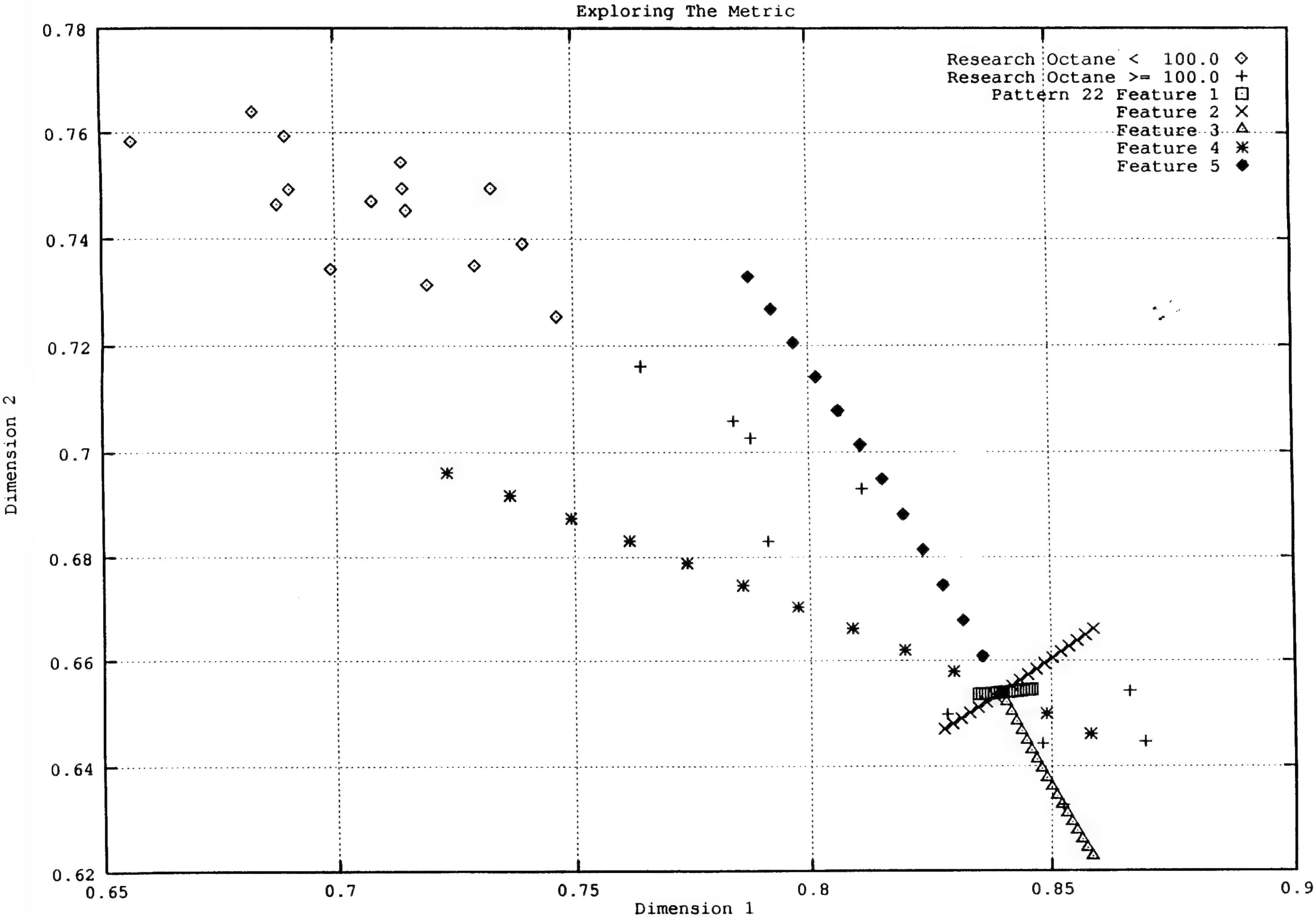


Exploring The Metric
Research Octane < 100.0
Research Octane >= 100.0
Pattern 20 Feature 1
Feature 2
Feature 3
Feature 4
Feature 5
Dimension 1
Dimension 2
0.65
0.7
0.75
0.8
0.85
0.9
0.62
0.64
0.66
0.68
0.7
0.72
0.74
0.76
0.78



Exploring The Metric
Research Octane < 100.0
Research Octane >= 100.0
Pattern 21 Feature 1
Feature 2
Feature 3
Feature 4
Feature 5
Dimension 1
Dimension 2
0.65
0.7
0.75
0.8
0.85
0.9
0.62
0.64
0.66
0.68
0.7
0.72
0.74
0.76
0.78

220

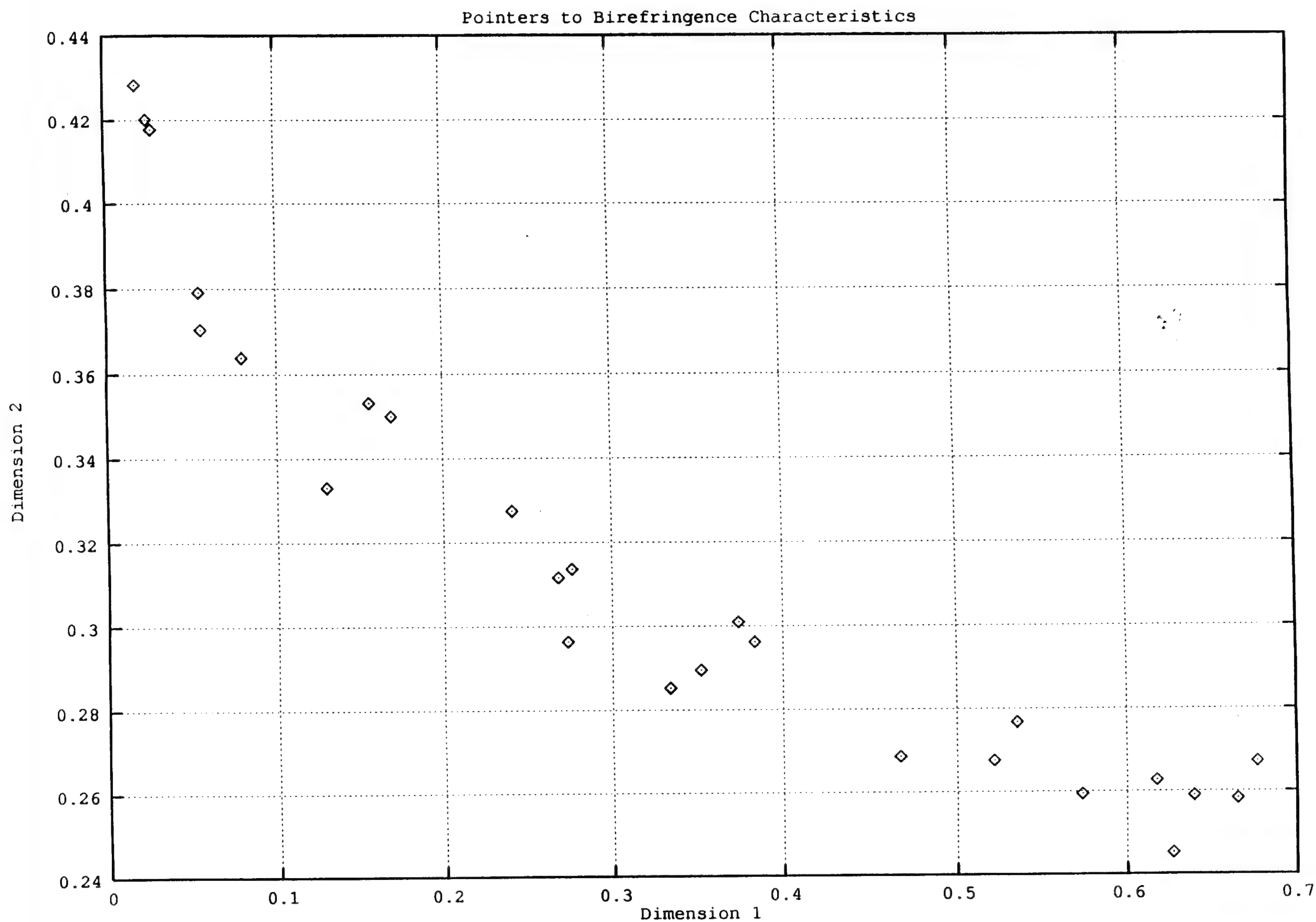


Exploring The Metric
Research Octane < 100.0
Research Octane >= 100.0
Pattern 22 Feature 1
Feature 2
Feature 3
Feature 4
Feature 5
Dimension 1
Dimension 2
0.65
0.7
0.75
0.8
0.85
0.9
0.62
0.64
0.66
0.68
0.7
0.72
0.74
0.76
0.78

L16

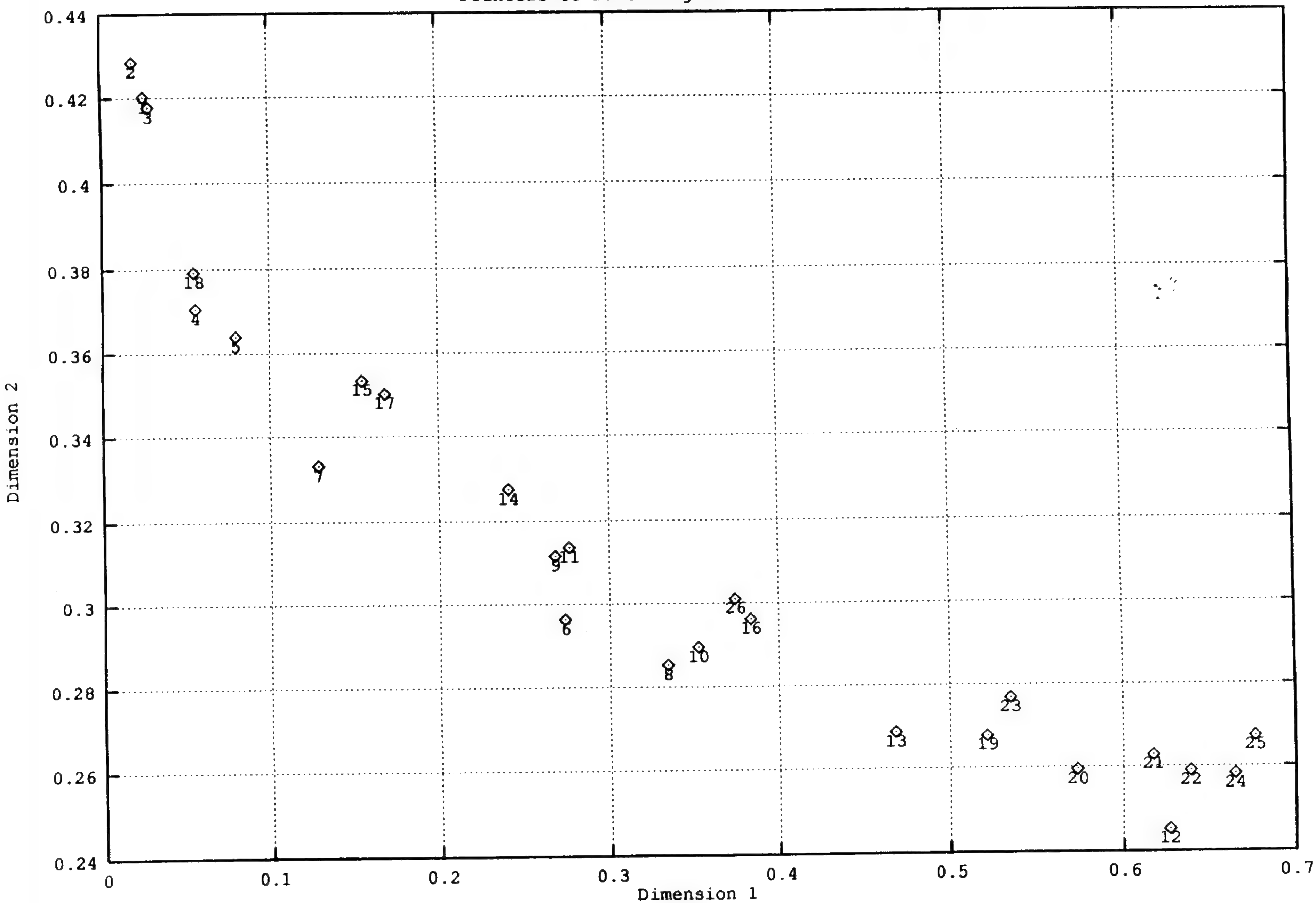
No.	Compounds	u	a	c	c/a	Gap
1	AgGaS2	0.28	5.75722	10.3036	1.790	2.55
2	AgAlS2	0.3	5.73	10.3	1.798	3.13
3	AgGaSe2	0.27	5.755	10.28	1.786	1.8
4	CdSiAs2	0.298	5.884	10.882	1.849	1.55
5	CdGeP2	0.2839	5.738	10.765	1.876	1.72
6	AgAlTe2	0.26	6.296	11.83	1.879	2.25
7	CdGeAs2	0.278	5.9432	11.2163	1.887	0.6
8	AgGaTe2	0.26	6.3197	11.9843	1.896	1.1
9	AgLnTe2	0.25	5.836	11.1789	1.916	1.9
10	CdSnP2	0.265	5.9	11.518	1.952	1.7
11	CuAlSe2	0.26	5.6103	10.982	1.957	2.6
12	AgLnSe2	0.25	6.455	12.644	1.959	0.96
13	CdSnAs2	0.262	6.09	11.94	1.961	0.26
14	ZnGeP2	0.25816	5.46	10.71	1.962	2.34
15	CuAlS2	0.27	5.31	10.42	1.962	3.35
16	ZnGeAs2	0.25	5.66	11.154	1.971	0.75
17	CuFeS2	0.27	5.289	10.423	1.971	0.53
18	AgAlSe2	0.27	5.95	10.75	1.807	2.6
19	CuAlTe2	0.25	5.964	11.78	1.975	2.06
20	CuGaTe2	0.25	6.013	11.934	1.985	1.24
21	CuTiSe2	0.25	5.832	11.63	1.994	1.07
22	ZnSnAs2	0.231	5.851	11.702	2.000	0.65
23	ZnSnP2	0.238	5.65	11.3	2.000	1.66
24	ZnLnSe2	0.224	5.784	11.614	2.008	0.95
25	CuLnS2	0.2	5.5228	11.1321	2.106	1.54
26	CuGaS2	0.25	5.555	11.0036	1.981	1.71

Table 2: Birefringence Data

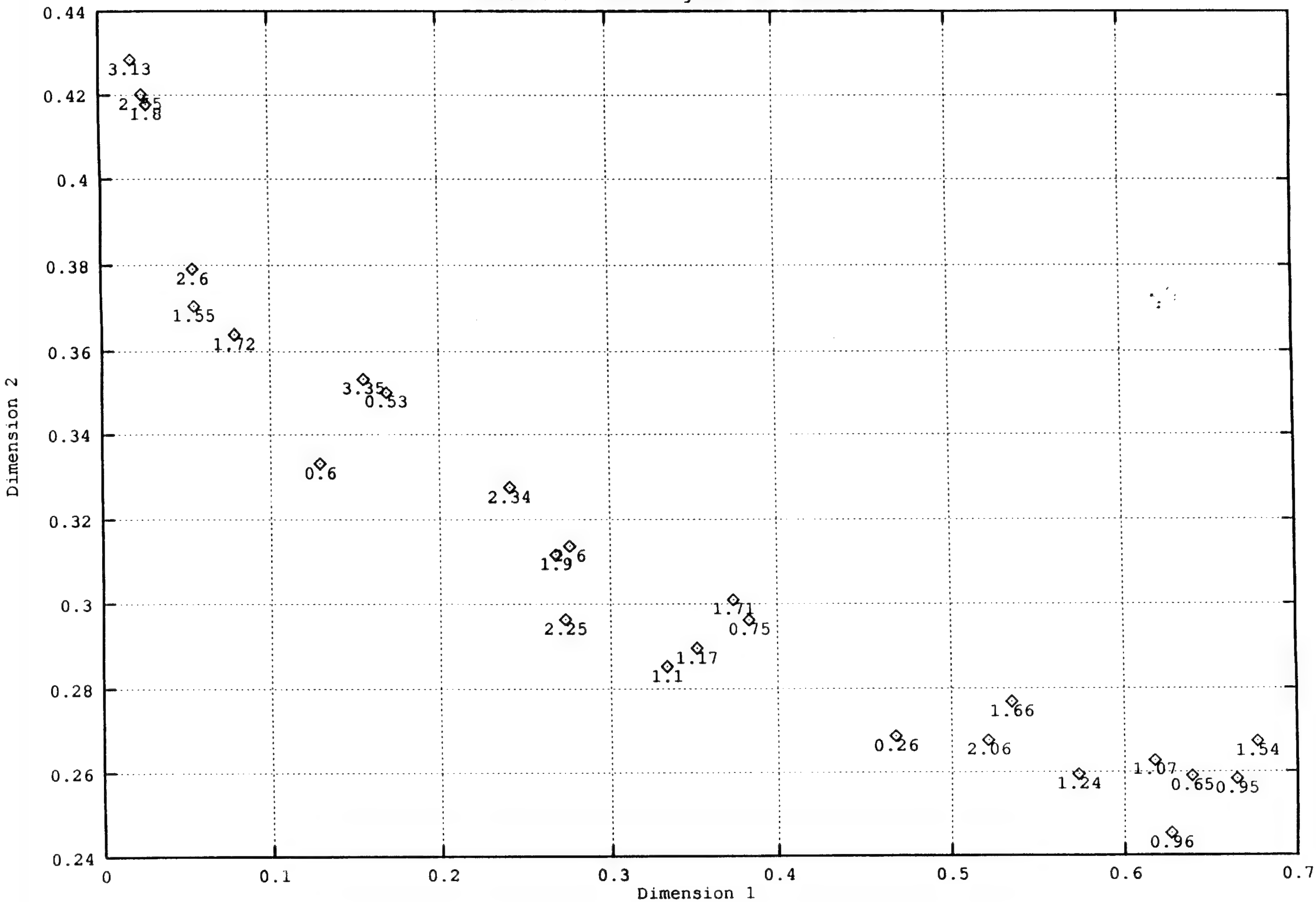

Pointers to Birefringence Characteristics
0.44
0.42
0.4
0.38
0.36
0.34
0.32
0.3
0.28
0.26
0.24
Dimension 2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Dimension 1

616

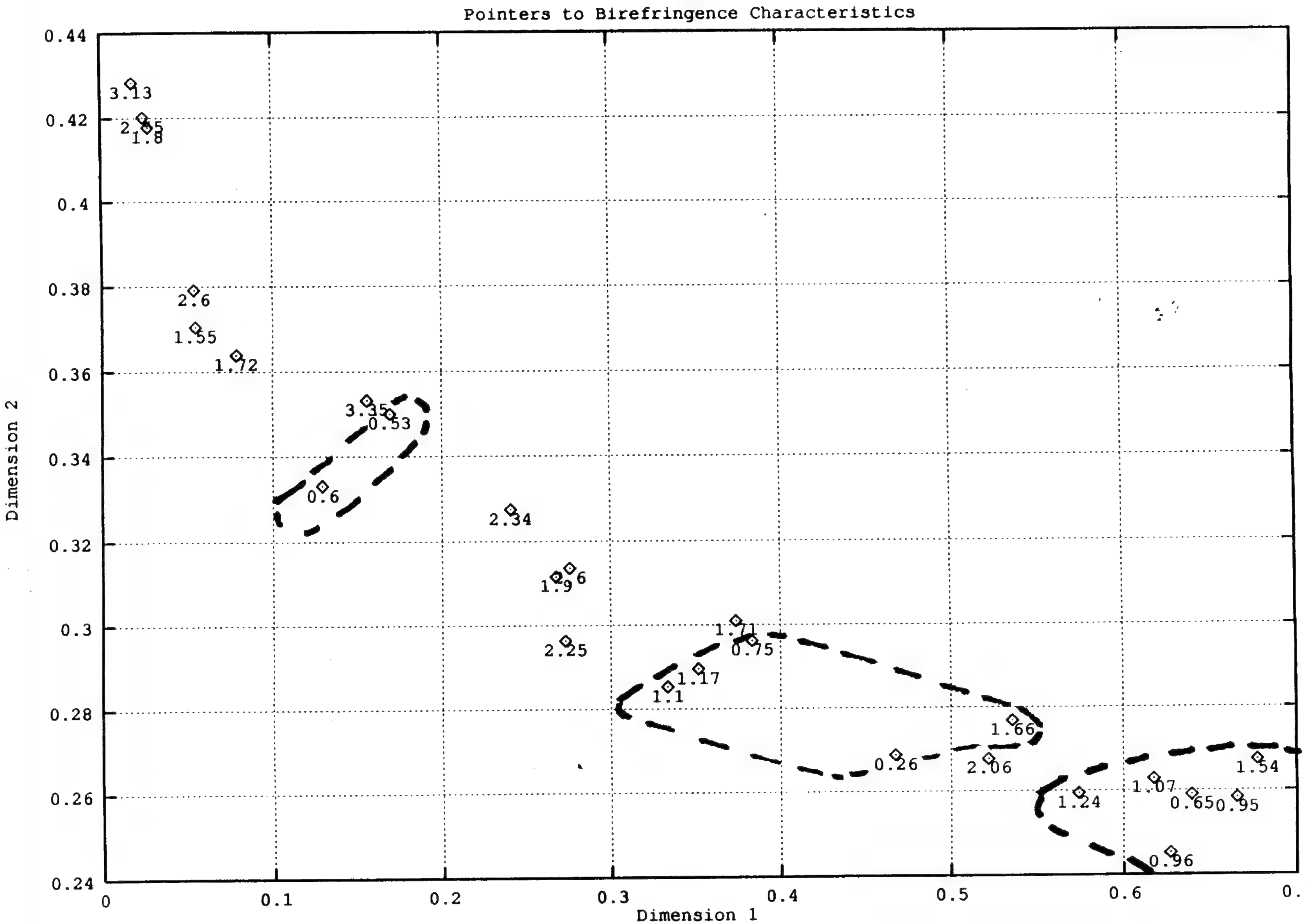
Pointers to Birefringence Characteristics



Pointers to Birefringence Characteristics



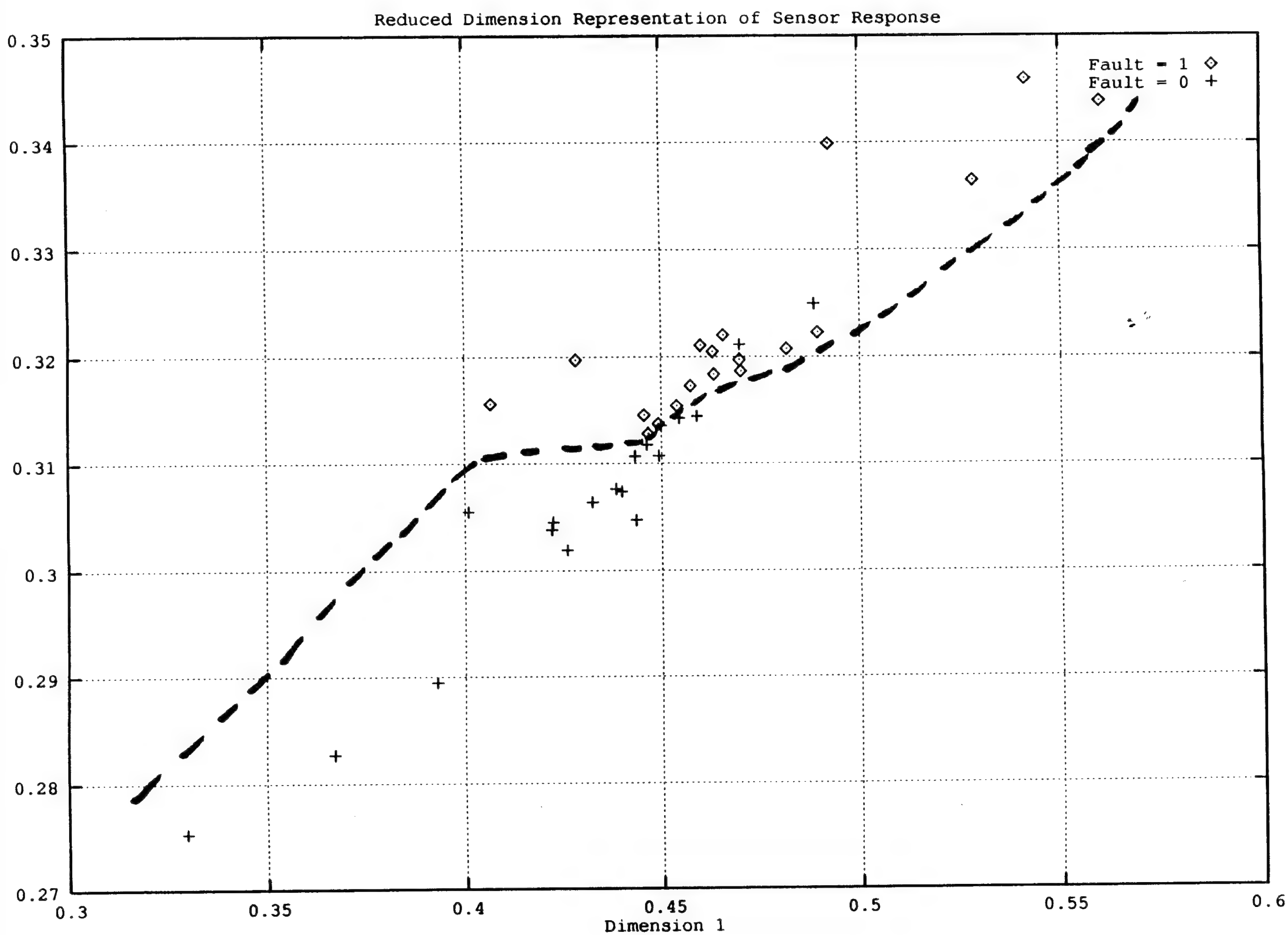
221


Pointers to Birefringence Characteristics
Dimension 1
Dimension 2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.24
0.26
0.28
0.3
0.32
0.34
0.36
0.38
0.4
0.42
0.44
3.13
1.8
2.6
1.55
1.72
0.6
0.53
2.34
1.9
2.25
1.17
1.1
0.75
1.66
0.26
2.06
1.24
1.07
0.65
0.95
1.54
0.96

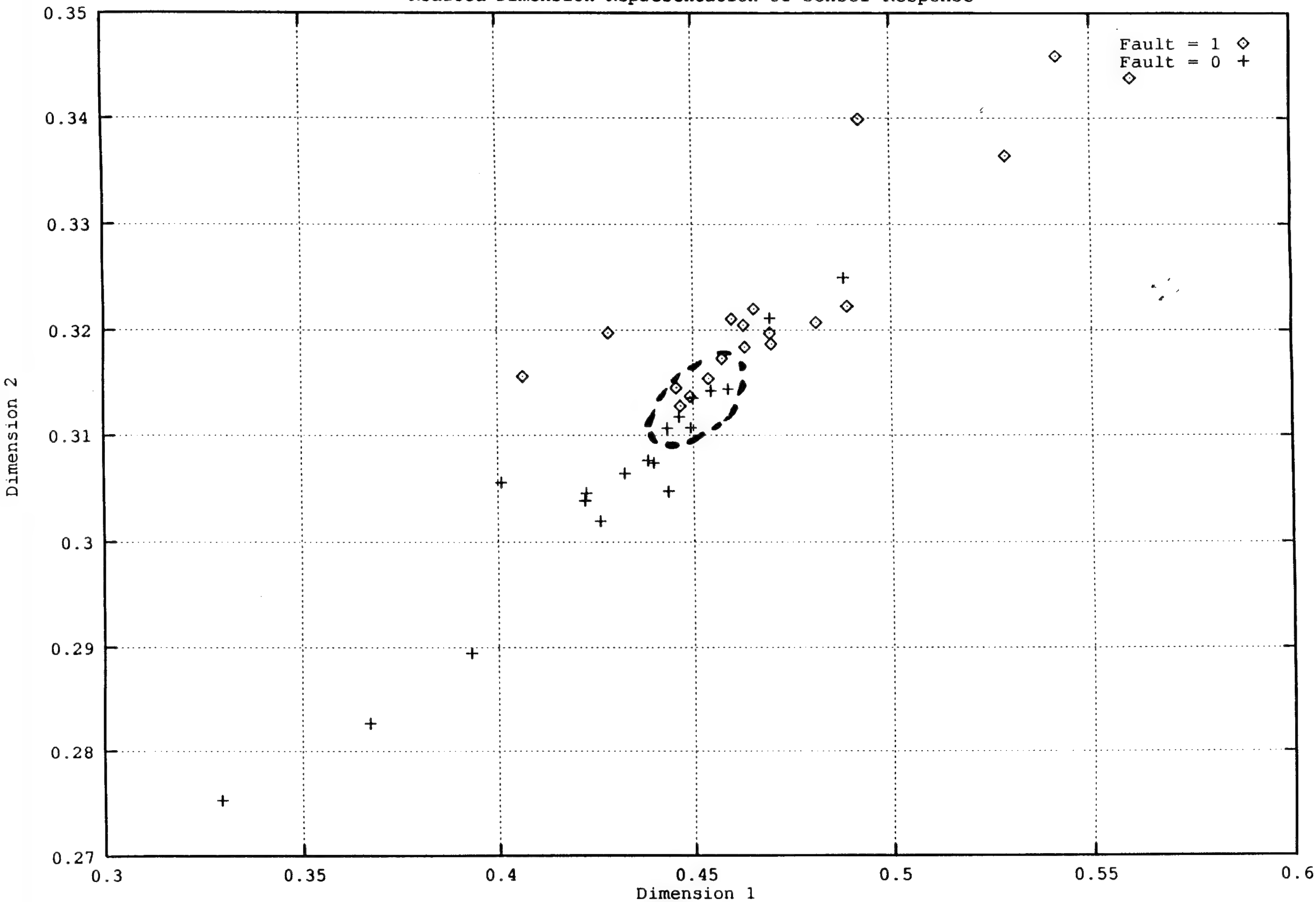
222

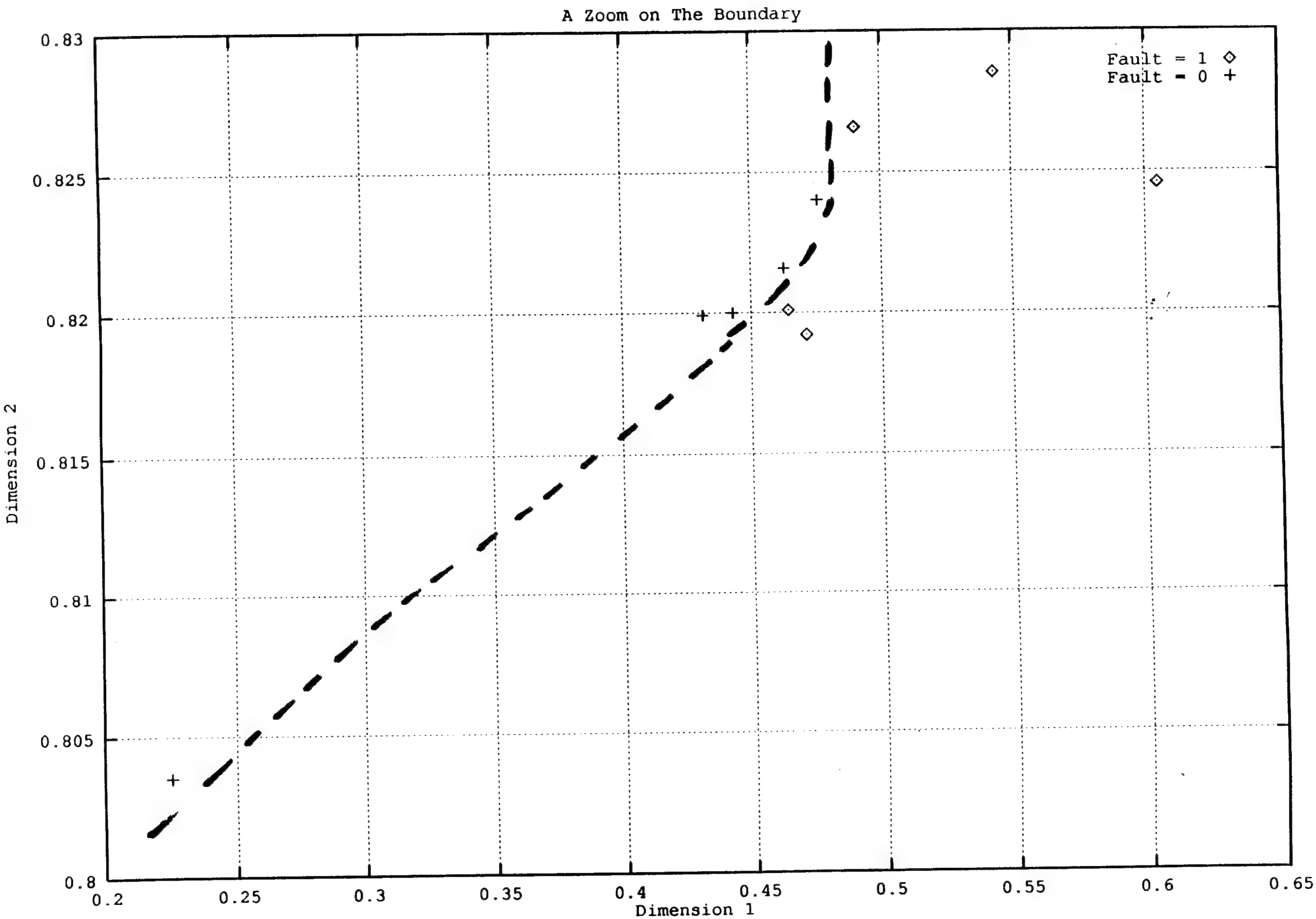
No.	t:1-5	t:6-11	t:12-17	t:18-23	t:24-29	Fault
1	0.65190	0.13019	0.31398	0.69901	0.30067	0.00000
2	0.27577	0.56790	0.24946	0.61443	0.70156	1.00000
3	0.86528	0.30303	0.10538	0.56716	0.58797	0.00000
4	0.15642	0.83277	0.58065	0.37313	0.58352	1.00000
5	0.82369	0.27834	0.24731	0.67413	0.90200	0.00000
6	0.35353	0.67116	0.16559	0.65920	0.82405	1.00000
7	0.40958	0.35241	0.41290	0.73881	0.70601	0.00000
8	0.35443	0.33782	0.55054	0.70647	0.71269	1.00000
9	0.54702	0.5735	0.59355	0.67413	0.72606	0.00000
10	0.34177	0.60718	0.79355	0.79851	0.64588	1.00000
11	0.47920	0.65208	0.67312	0.83582	0.74833	0.00000
12	0.35353	0.57800	0.94409	0.95025	0.74610	1.00000
13	0.47197	0.32099	0.36559	0.58209	0.52561	0.00000
14	0.36528	0.39843	0.44731	0.61940	0.55457	1.00000
15	0.44123	0.29854	0.34624	0.57711	0.55457	0.00000
16	0.35805	0.35354	0.42150	0.59701	0.56793	1.00000
17	0.49005	0.32997	0.41505	0.72139	0.67929	0.00000
18	0.31284	0.43547	0.43656	0.72388	0.70601	1.00000
19	0.43309	0.31874	0.39785	0.71642	0.73497	0.00000
20	0.34991	0.36251	0.44946	0.71144	0.73051	1.00000
21	0.46745	0.26936	0.40860	0.69652	0.72160	0.00000
22	0.35262	0.37261	0.42366	0.70398	0.70601	1.00000
23	0.59042	0.25253	0.48602	0.78358	0.82628	0.00000
24	0.38427	0.37486	0.48172	0.79851	0.80401	1.00000
25	0.38156	0.19753	0.40645	0.63930	0.83296	0.00000
26	0.34810	0.52189	0.44516	0.68906	0.72160	1.00000
27	0.75769	0.91134	0.44301	0.61194	0.51225	0.00000
28	0.41863	1.00000	1.00000	0.59453	0.49220	1.00000
29	0.50723	0.36364	0.40645	0.68159	0.71715	0.00000
30	0.34991	0.47250	0.45806	0.70149	0.70156	1.00000
31	0.54069	0.24691	0.38279	0.70647	0.73051	0.00000
32	0.38788	0.40404	0.38710	0.70149	0.72383	1.00000
33	0.41320	0.32660	0.41075	0.68408	0.71715	0.00000
34	0.34991	0.34007	0.49247	0.68906	0.70379	1.00000
35	0.39873	0.35354	0.44516	0.68906	0.69710	0.00000
36	0.33906	0.32323	0.58065	0.70149	0.69710	1.00000
37	0.29747	0.26824	0.42366	0.74378	0.85746	0.00000
38	0.30561	0.21886	0.36129	0.59950	0.67038	1.00000

Table 3: Typical Sensor Data


Reduced Dimension Representation of Sensor Response
Fault = 1
Fault = 0
Dimension 1
Dimension 2
0.3
0.35
0.4
0.45
0.5
0.55
0.6
0.27
0.28
0.29
0.3
0.31
0.32
0.33
0.34
0.35

Reduced Dimension Representation of Sensor Response




A Zoom on The Boundary
Fault = 1 ◇
Fault = 0 +
Dimension 2
0.83
0.825
0.82
0.815
0.81
0.805
0.8
0.2
0.25
0.3
0.35
0.4
0.45
0.5
0.55
0.6
0.65
Dimension 1

Materials Design Using Pyramidal Networks

by A. Jackson
Wright Laboratory

Abstract

Created by V. Gladun (Ukranian Academy of Sciences) in the late sixties, pyramidal nets are a means for classifying objects into a directed graph that allows rules to be written relating the objects (instances) and their attributes (data). For materials design, pyramidal nets allow prediction of materials and properties to be done using empirical data. The net consists of attributes, nodes representing collections of attributes, and the objects. Once the net is created, special nodes can be identified that characterize sets of objects and which allow logical conjunctions and disjunctions to be formulated that provide insight into the relationships among the objects or the attributes. Two examples are presented. The first deals with a simple set of properties for materials that illustrate the generation algorithm. The second example is for a set of chalcopyrite compounds and illustrates the utility of the pyramidal net in associating properties to compounds or compounds to properties.

Design of Materials is an intensive effort requiring contributions from many individuals with a wide range of backgrounds. The time spent designing a new material can be considerable, ranging from a few years to many. In contrast to design of objects, design of materials has not received as much attention, because the process of materials design is rather complex and the foundations are only now being explored and articulated. A number of approaches are being considered, including neural nets of various varieties, genetic algorithms and optimization techniques, and various forms of logic systems, for example fuzzy logic, rough sets, and pyramidal nets, the subject of this presentation. Created by V. Gladun in the late sixties, pyramidal nets are a means for classifying objects into a directed graph that allows rules to be written relating the objects and their attributes. For materials design, pyramidal nets allow prediction of materials and properties to be done using empirical data. The net consists of attributes, nodes representing collections of attributes, and the objects. Once the net is created, special nodes can be identified that characterize sets of objects and which allow logical conjunctions and disjunctions to be formulated that provide insight into the relationships among the objects or the attributes.

228

Materials Design Using Pyramidal Networks

A. G. Jackson
Wright Laboratory

Electronic Prototyping Review
Wright State University
May 23-24, 1995

Crystal Engineering

Thin Films

- Coatings
- Sensors
- Devices

Problem - Drivers

- Property Prediction
- Compound Prediction
- Discovery of New Information

Development of Materials Design Methods

Neural Nets
Genetic Algorithms
Fuzzy Logic
Rough Sets
Pyramidal Nets

Gladun Algorithm

Sources

Books by Gladun:
- *Processes of the Formation of New Knowledge*
- *Planning Decisions*
- *Heuristics in Complex Media*

Reports by or edited by LeClair & Jackson:
- *Trip Report: Window on Europe*
- *Innovations in Materials Design, Part 1: Design of Inorganic Compounds*

what is it . . . . .

The Algorithm

0. $B' = \{0\}$;
1. index $i = 0$;
2. index $i = i + 1$;
3. If for every $\tilde{N}_j$ in B' for which N_j is a subset of or equal to set K_i is true, then generate the function $Q_1(K_i, N_j)$, where
   $Q_1(K_i, N_j) = K_i := K_i[N_j]$;
4. If for every $\tilde{N}_j$ in B' for which $|N_j \cap K_i|>1$ is true, then generate the function $Q_2(K_i, N_j)$, where
   $Q_2(K_i, N_j) = (\tilde{N}_t := K_i \cap N_j;\ \tilde{N}_j := N_j[N_t];\ \tilde{}K_i = K_i[N_t];\ \tilde{N}_t$ in $B \backslash B')$
5. If $|K_i|>1$ is true, then execute $Q_3(K_i)$, where
   $Q_3(K_i) = (\tilde{N}_j := K_i(\tilde{N}_j$ in $B \backslash B')$;
6. If $i \neq n$, then go to step 2.

$B = \{\tilde{N}j\}$ is the set of names of the objects; B' Ã B; N_j Ã K_i .

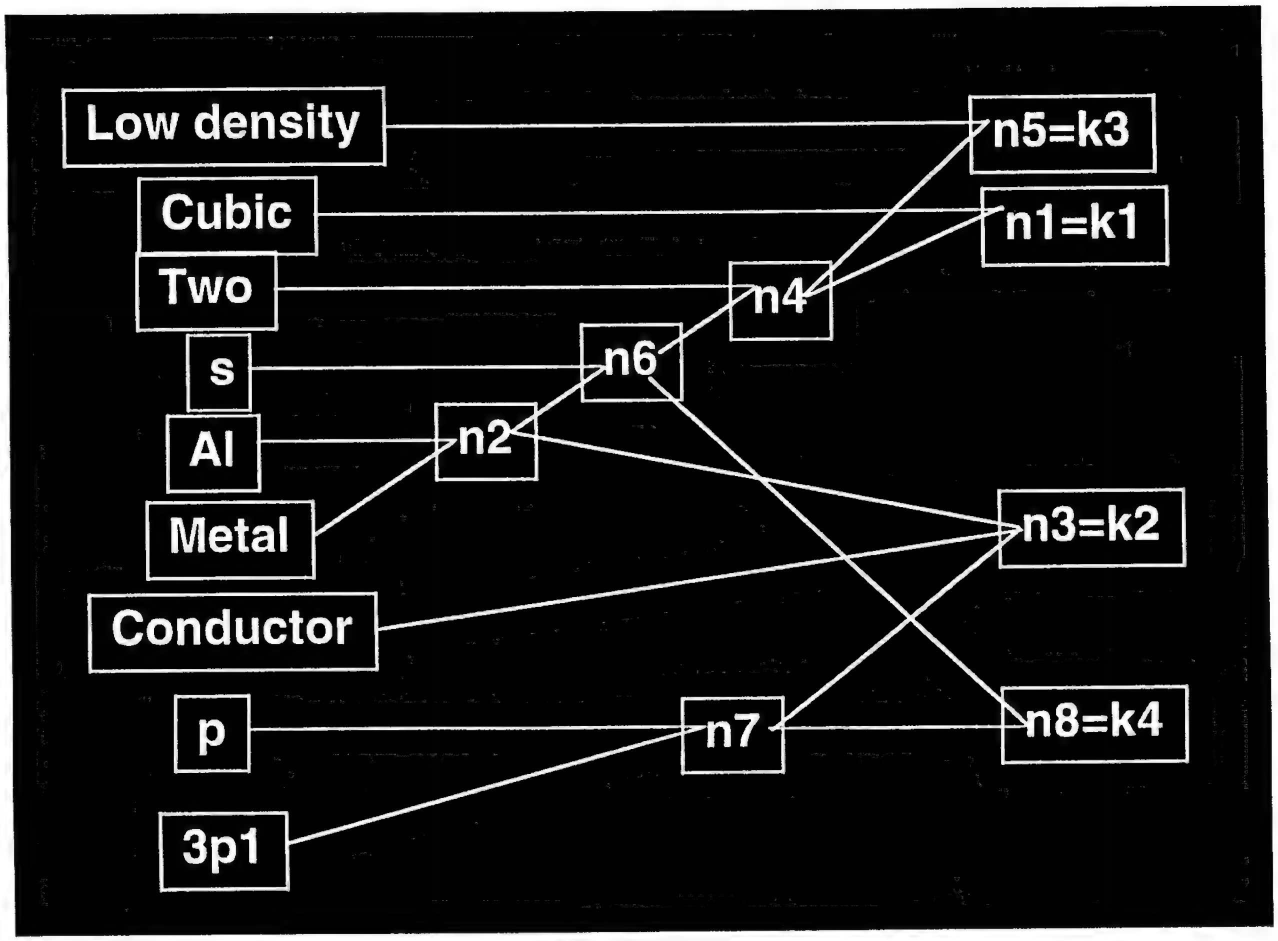
Example 1

Attributes:	Values:
name	Al
type of material	metal
electron shell symbols	s, p
electrical conductivity	conductor
density	lowdensity
electron configuration	2,3p1
structure	cubic

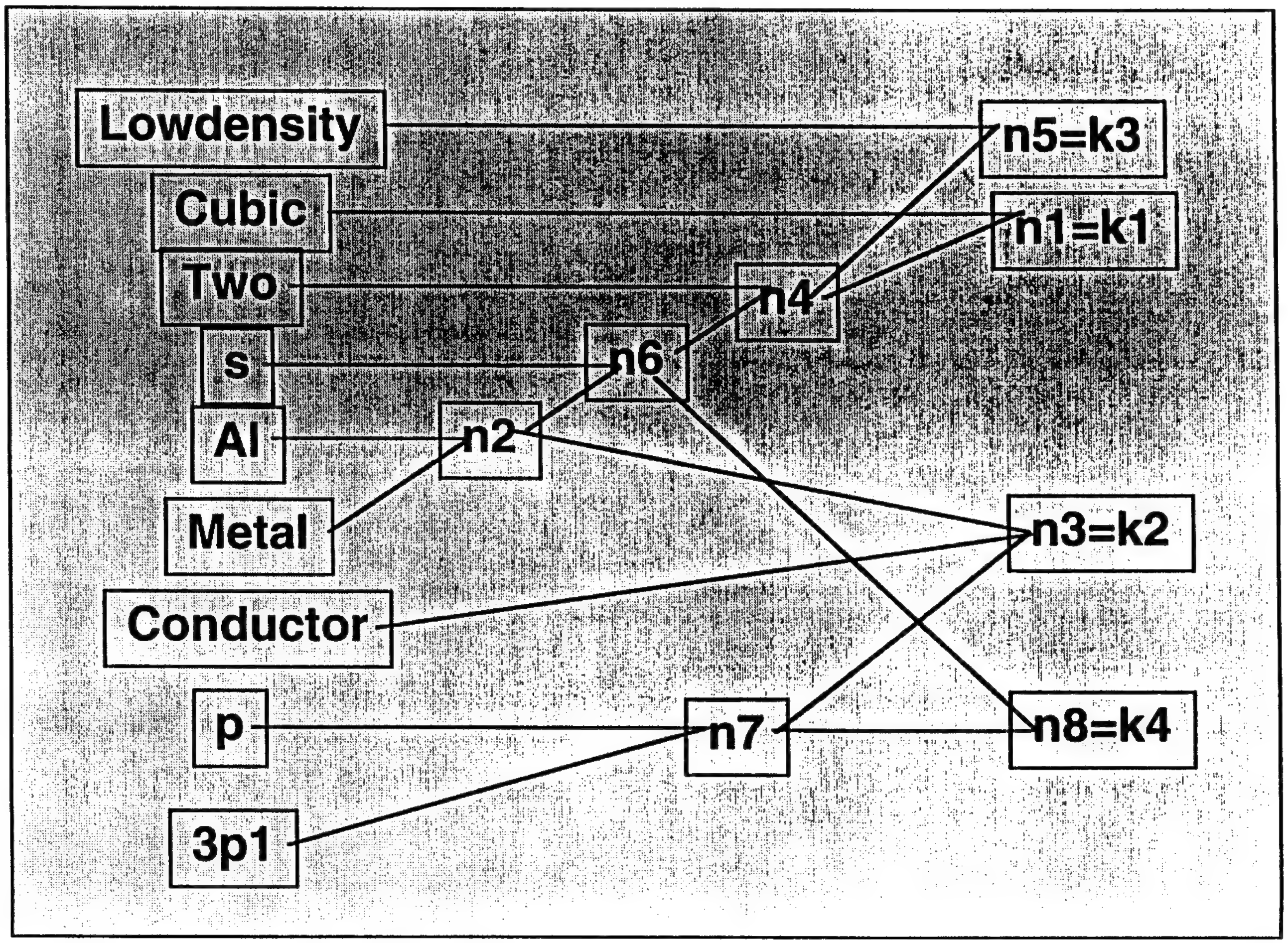
Objects

K1 = {Al, metal,s,two, cubic}
K2 = {Al, metal,p,three-p-one,conductor}
K3 = {Al,metal,s,two,lowdensity}
K4 = {Al,metal,s,p,three-p-one}

Pyramidal Net



Pyramidal Net



Logical Forms of Nodes

Primary Sub-Objects

n2 = (Al and Metal)

n7 = (p and 3p1)

Sub-Objects

n6 = (n2 and s and not n3)

n4 = (n6 and two and not n8)

Objects

n8 = (n7 and n6 and not n4)

n3 = (n7 and conductor and n2 and not n8 and not n6)

n1 = (cubic and n4 and not n5)

n5 = (lowdensity and n4 and not n1)

Example 2

Chalcopyrite Compounds

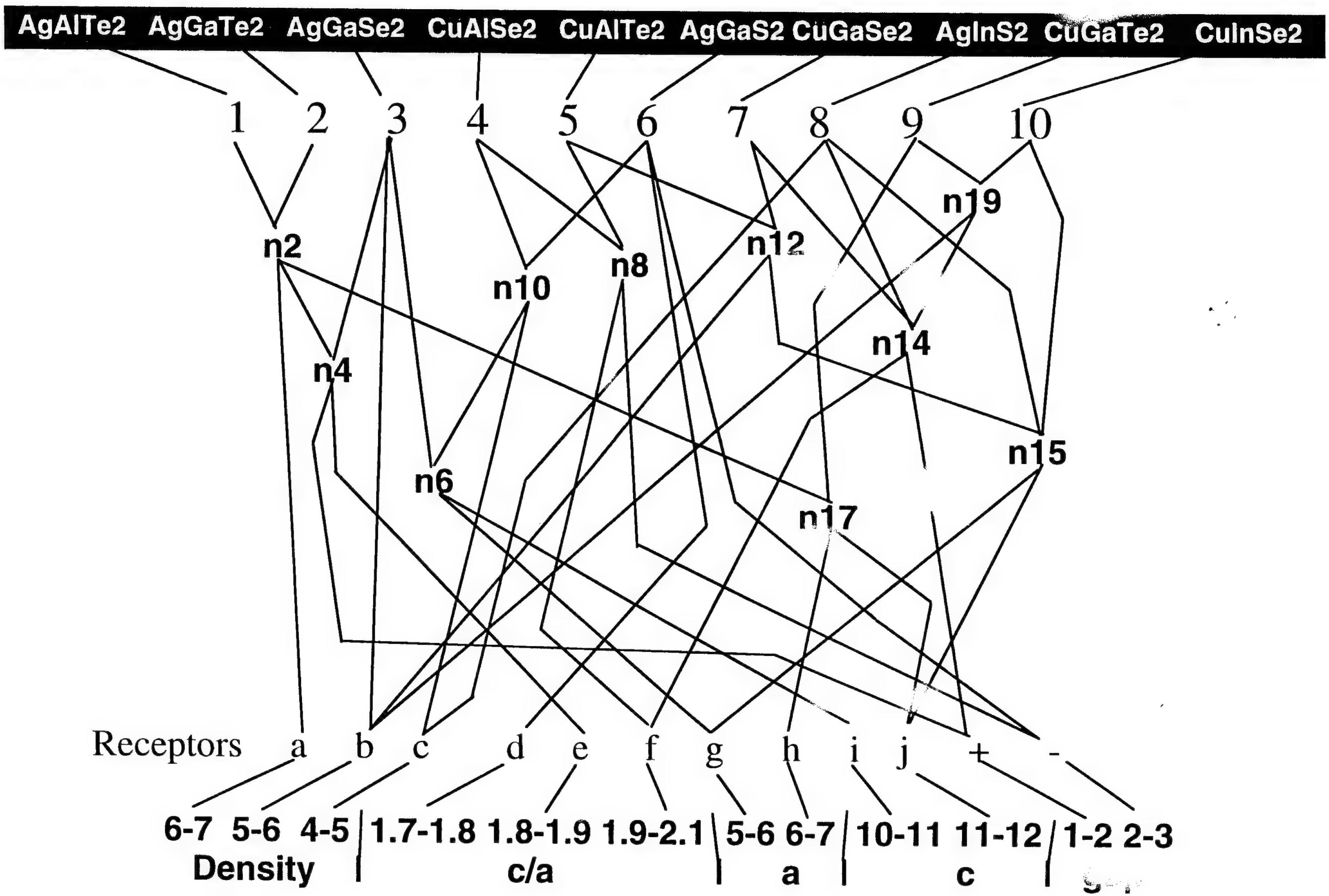
Electro-Optical and Structural Data

Data for 10 Chalcopyrite Compounds

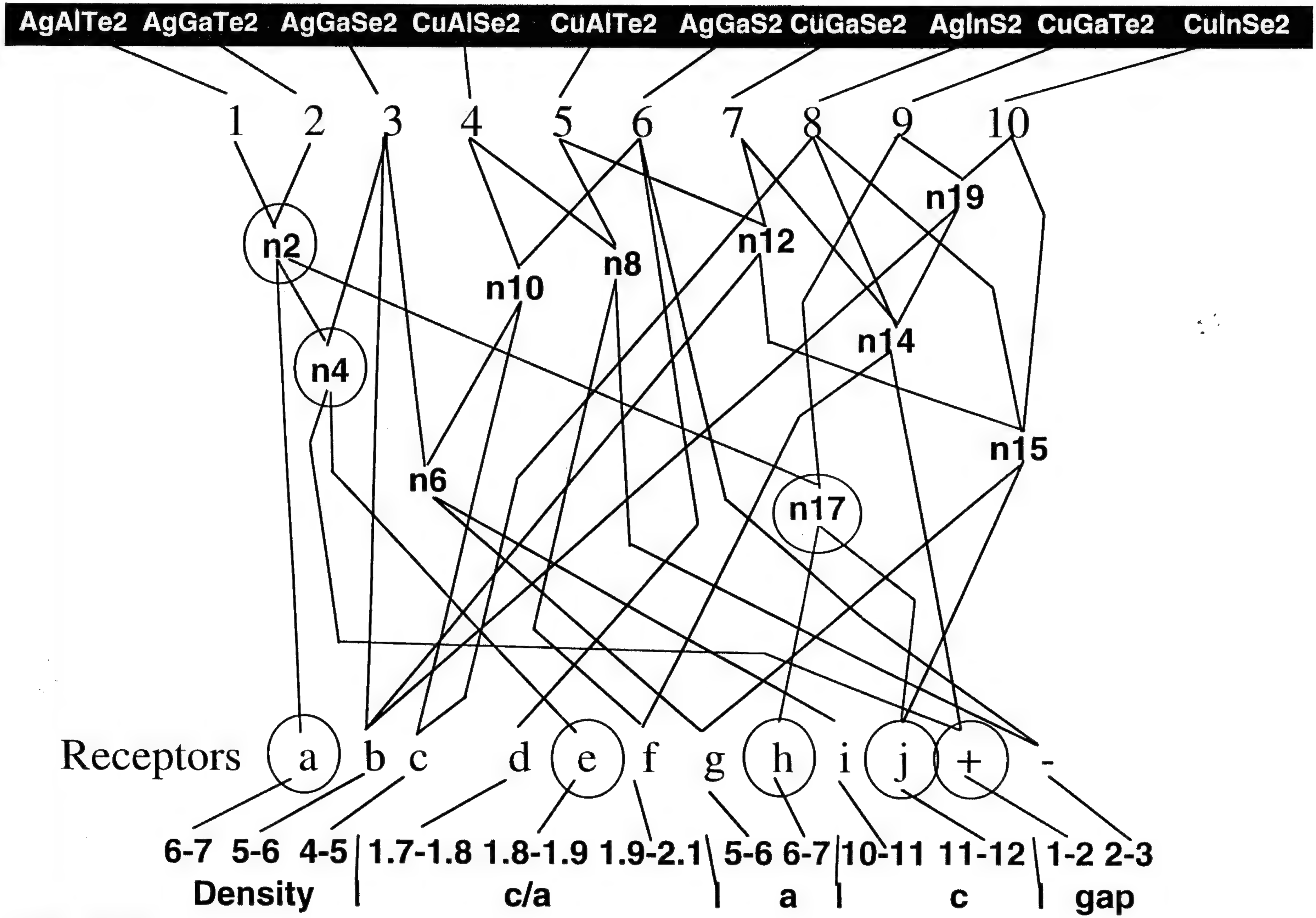
Compound	Density	c/aRatio	aParar	cPara	Gap
AgAlTe2	6.172	1.878	6.303	11.842	1.415
AgGaTe2	6.254	1.897	6.297	11.956	1.175
AgGaSe2	5.759	1.816	5.98	10.865	1.806
CuAlSe2	4.697	1.948	5.607	10.929	2.656
CuAlTe2	5.492	1.973	5.967	11.792	2.06
AgGaS2	4.66	1.787	5.751	10.283	2.637
CuGaSe2	5.505	1.962	5.606	11.006	1.693
AgInS2	4.984	1.917	5.832	11.185	1.942
CuGaTe2	5.93	1.985	6.003	11.795	1.236
CuInSe2	5.71	2.006	5.781	11.597	1.006

Definitions for Bins

Density	a	6-7
	b	5-6
	c	4-5
c/a	d	1.7-1.8
	e	1.8-1.9
	f	1.9-2.1
a	g	5-6
	h	6-7
c	i	10-11
	j	11-12
gap	+	1-2
	-	2-3


AgAlTe2 AgGaTe2 AgGaSe2 CuAlSe2 CuAlTe2 AgGaS2 CuGaSe2 AgInS2 CuGaTe2 CuInSe2
1 2 3 4 5 6 7 8 9 10
n2 n4 n6 n8 n10 n12 n14 n15 n17 n19
Receptors a b c d e f g h i j + -
6-7 5-6 4-5 1.7-1.8 1.8-1.9 1.9-2.1 5-6 6-7 10-11 11-12 1-2 2-3
Density
c/a
a
c

039



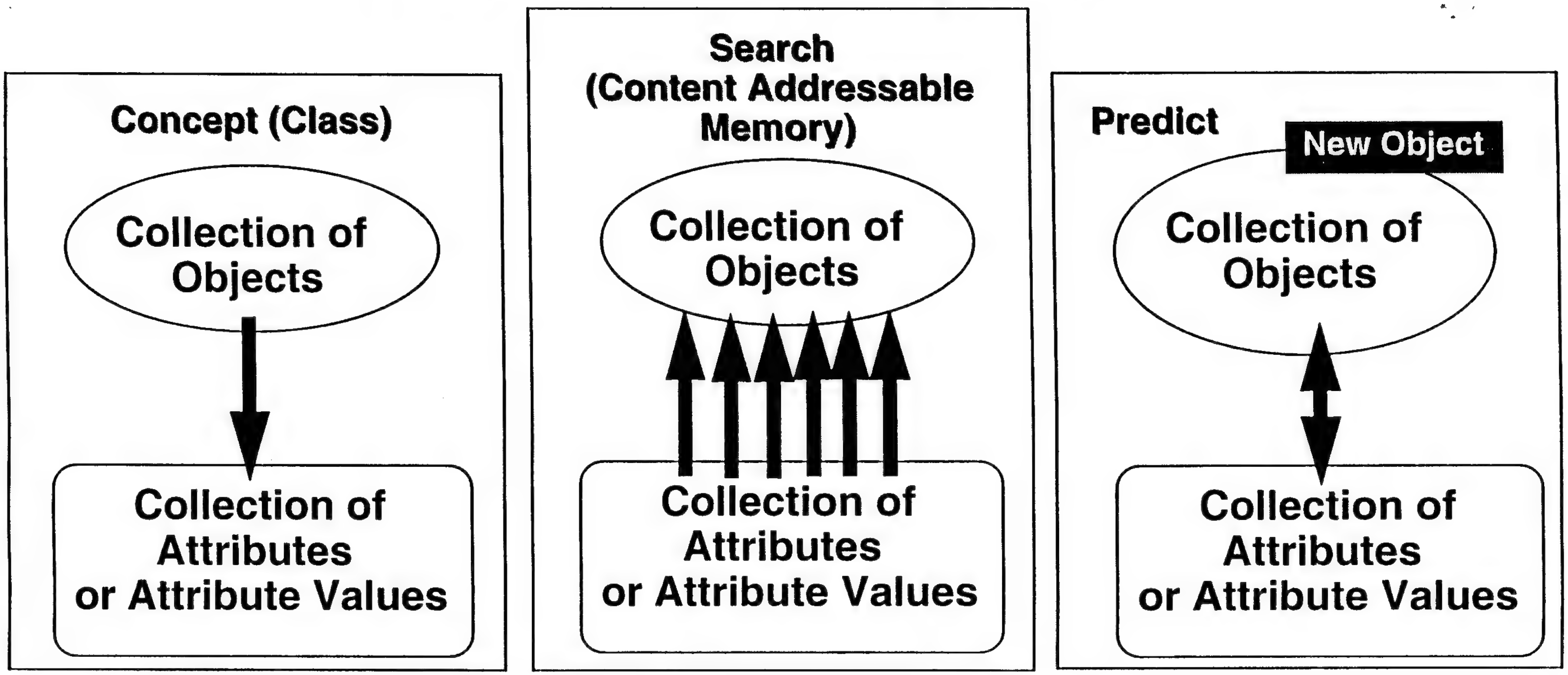
AgAlTe2 AgGaTe2 AgGaSe2 CuAlSe2 CuAlTe2 AgGaS2 CuGaSe2 AgInS2 CuGaTe2 CuInSe2
1 2 3 4 5 6 7 8 9 10
n2
n4
n6
n8
n10
n12
n14
n15
n17
n19
Receptors
a b c d e f g h i j + -
6-7 5-6 4-5
Density
1.7-1.8 1.8-1.9 1.9-2.1
c/a
5-6 6-7
a
10-11 11-12
c
1-2 2-3
gap

240

What is Produced?

Associative Net

Concept Formation
Search Simplification
Prediction Capability



241

Questions

- What patterns and information can be discerned in the birefringence data set?

- What displays are appropriate for algorithm outputs?
- How is the algorithm used with other discovery methods?
- What foundational connections are there among pyramidal nets, rough sets, neural nets, and genetic algorithms?
- What materials problems are best suited to these methods?

Implementation of the Algorithm:

CONFOR

Created by

V. Gladun and N. Vaschenko,
Institute of Informatiks
Kiev, Ukraine

A Rapid Supervised Learning Neural Network for Function Interpolation and Approximation

by C. L. Philip Chen, Senior Member, IEEE
Department of Computer Science and Engineering
Wright State University, Dayton, OH 45435
and Visiting Research Scientist
Wright Laboratory, WL/MLIM
Wright-Patterson Air Force Base, OH 45433

Abstract

A unique implementation of the Functional-Link neural network is addressed which focuses on an instant learning algorithm to rapidly compute the weights for supervised learning. For an n-dimensional, N-pattern training set, with a constant bias, a maximum of N-r-1 enhancement nodes are required to learn the mapping within a given precision (where r is the rank, usually the dimension, of the input training set). Based on linear algebra theorems, a neural network which is similar to the architecture of the Functional-Link neural network is proposed. In this way the learning process is equivalent to the process of finding an optimal weight matrix. Since the neural network is a flat network, the optimal weights can be solved easily through linear least squares computation. For off-line training, the proposed network and algorithm is able to achieve "one-shot" training as opposed to the more iterative training algorithms in the literature. An on-line iterative training algorithm is also presented. Similar to most of back propagation type of learning algorithms, the given algorithm also interpolates the training data. To eliminate outlier data which may appear in some erroneous training data, a robust weighted least squares method is proposed. The robust weighted least squares learning algorithm can eliminate outlier samples and the algorithm approximates the training data rather than interpolates them. Several experiments show very promising results.

A Rapid Supervised Learning Neural Network for Function Interpolation and Approximation

C. L. Philip Chen, Senior Member, IEEE

Department of Computer Science and Engineering
Wright State University, Dayton, OH 45435
Visiting Research Scientist
Wright Laboratory, WL/MLIM
Wright-Patterson Air Force Base, OH 45433

ABSTRACT

This talk presents a unique implementation of the Functional-Link neural network and an instant learning algorithm that rapidly decides the weights of the supervised learning. For an n dimensional N-pattern training set, with a constant bias, a maximum of $N-r-1$ enhancement nodes are required to learn the mapping within a given precision (where r is the rank, usually the dimension, of the input patterns). Based on linear algebra theorems, a neural network which is similar to the architecture of the Functional-Link neural network was proposed. In this way the learning process is equivalent to the process of finding an optimal weight matrix. Since the neural network is a flat network, the optimal weight can be solved easily through linear least squares computation. For off-line training, the proposed network and algorithm is able to achieve "one-shot" training as opposed to most iterative training algorithms in the literature. An on-line iterative training algorithm is also presented. Similar to most of back propagation type of learning algorithms, the given algorithm also interpolates the training data. To eliminate outlier data which may appear in some erroneous training data, a robust weighted least squares method is proposed. The robust weighted least squares learning algorithm can eliminate outlier samples and the algorithm approximates the training data rather than interpolates them. Several experiments show very promising result.

A Rapid Supervised Learning Neural Network for Function Interpolation and Approximation

C. L. Philip Chen, Senior Member, IEEE

Department of Computer Science and Engineering

Wright State University, Dayton, OH 45435

Visiting Research Scientist

Wright Laboratory, WL/MLIM

Wright-Patterson Air Force Base, OH 45433

Acknowledgement

Air Force Contract F33610-D-5964, Wright Laboratory,

Wright-Patterson AFB, Ohio and AFOSR F49620-94-0277

246

Outlines:

- Least-Square Neural Networks

  - ☐ A variation of the Functional-Link Net

- One Shot Learning for off-line training

- Iterative Learning for on-line training

- Function Interpolation

- Function Approximation

- Examples

Consider a mapping from an input space , R^n, to an output space, R^m.

$$\textbf{Output} = f\,(\textbf{ input })$$

- An n dimensional N-pattern input training set
- An m dimensional N-pattern output training set

☐ Statistical Regression:

Linear and Nonlinear — Closed form solution

☐ Neural Networks Approach

Training — Analytic solution

• Neural Networks Approach

Output = σ $\cdots$ σ (σ (Input * Weights)) * Weights) $\cdots$ Weights

where σ is a nonlinear function such as

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

$$\text{sigmoid}(x) = \frac{1}{1+e^{-x}}, \text{ or}$$

$$\text{radial basis function}(x) = exp\left(-\frac{||\mathbf{x}-\boldsymbol{\mu}||^2}{2\sigma^2}\right)$$

A Neural Net Figure here

- **Supervised Learning Approach**
  - Network Topology:

    Multi-layer vs Functional-link

  - Back propagation and its variants:

    Newton (Hessian), Conjugate Gradient etc.

  - Disadvantage of BP:

    Slow convergeness

    Retraining

Statistical Regression and Least Squares: A Review

$$
\begin{aligned}
\mathbf{y}_1 &= w_{01} + w_{11}\mathbf{x}_1 + \cdots + w_{n1}\mathbf{x}_n + \varepsilon_1 \\
\mathbf{y}_2 &= w_{02} + w_{12}\mathbf{x}_1 + \cdots + w_{n2}\mathbf{x}_n + \varepsilon_2 \\
\mathbf{y}_m &= w_{0m} + w_{1m}\mathbf{x}_1 + \cdots + w_{nm}\mathbf{x}_n + \varepsilon_m
\end{aligned}
\tag{2}
$$

where $\mathbf{y}_i$, $\mathbf{x}_i$, and ε_i are column vectors,

$$
\begin{aligned}
\mathbf{y}_i &= [y_{1i}, y_{2i}, \cdots, y_{Ni}]^T, \\
\mathbf{x}_i &= [x_{1i}, x_{2i}, \cdots, x_{Ni}]^T, \\
\varepsilon_i &= [\varepsilon_{1i}, \varepsilon_{2i}, \cdots, \varepsilon_{Ni}]^T.
\end{aligned}
$$

Expressed in the matrix form, the linear model of input and output is:

$$\mathbf{Y}_{(\mathbf{N}\times\mathbf{m})} = \mathbf{A}_{(\mathbf{N}\times(\mathbf{n}+1))}\mathbf{W}_{((\mathbf{n}+1)\times\mathbf{m})} + \varepsilon_{(\mathbf{N}\times\mathbf{m})} \tag{3}$$

In practice, the least squares computation is to minimize the error

$$\phi(W, \varepsilon) = \sum_k (\mathbf{x}_k \mathbf{w}_k^T - \mathbf{y}_k)^2 , \tag{4}$$

where $\mathbf{w}_k$ and ε_k can be solved by making $\phi(W,\varepsilon)$ a minimum, i.e.,

$$\frac{\partial\phi}{\partial W} = 0 \text{ and } \frac{\partial\phi}{\partial\varepsilon} = 0.$$

In general, a linear family of basis functions,
$g_0, g_1, \cdots, g_k$ such that

$$y_j = \sum_j c_j g_j(x) \tag{5}$$

in which the basis functions are known and held fixed. The coefficients, c_j, are to be determined according to the principle of least squares.

In other words, we want to minimize the expression

$$\phi(c_0, c_1, \cdots, c_n) = \sum_i^N \left(\sum_j c_j g_j(x_i) - y_i\right)^2 \tag{6}$$

The necessary conditions for the minimum is

$$\frac{\partial \phi}{\partial c_i} = 0. \tag{7}$$

In practice, the above equations may be difficult to solve:

- The basis functions need to be not chosen carefully
- The basis functions should be linearly independent
- Well conditioned for numerical computation

Nonlinear regression model:

$$\mathbf{y}_i = \mathbf{f}(\mathbf{x}_i, \theta) + \varepsilon_i, \quad i = 1,2, \cdots, n, \tag{8}$$

where θ is a vector containing p parameters and $p < n$

- Least squares formulation for finding the coefficients
- Much more complicated

Linear Algebra Theorems

Theorem 1: Consider a system of equations $\mathbf{AW} = \mathbf{B}$, and denote the augmented matrix of the system $\mathbf{A_a} \triangleq [\, \mathbf{A} \mid \mathbf{B} \,]$. Assume that $\mathbf{W}$ is an $(n+1)\times m$ matrix.

(1) The system $\mathbf{AW} = \mathbf{B}$ has a solution if and only if $\mathbf{rank}(\mathbf{A}) = \mathbf{rank}(\mathbf{A_a})$.

(2) The system $\mathbf{AW} = \mathbf{B}$ has a unique solution if and only if $\mathbf{rank}(\mathbf{A}) = n+1 = \mathbf{rank}(\mathbf{A_a})$.

- **Supervised Learning:**

Find a weight matrix, **W**, given an A and a Y matrix.

☐ Neural Networks: One hidden-layer architecture

$$[\ \sigma(AW_1)\ ][\ W_2\ ] = B \triangleq \sigma^{-1}(Y)$$

where W_1 is the weight matrix from the input to the hidden nodes, and W_2 is the weight matrix from the hidden nodes to the output nodes.

- Increase number of hidden nodes, hopefully,

$\Rightarrow$ increase the rank of the matrix $[\ \sigma(AW_1)\ ]$, or

$$rank(\ [\ \sigma(AW_1)\ ]\ ) = rank(\ [\ \sigma(AW_1)\ \mid\ B\ ]\ ).$$

$\Rightarrow \mathbf{W} = A^{-1}\,B$ or $A^{+}\,B$

☐ Neural Networks: Another one hidden-layer architecture

$$[\, A \mid \sigma(AW_1)\,][\, W_2\,] = B \triangleq \sigma^{-1}(Y)$$

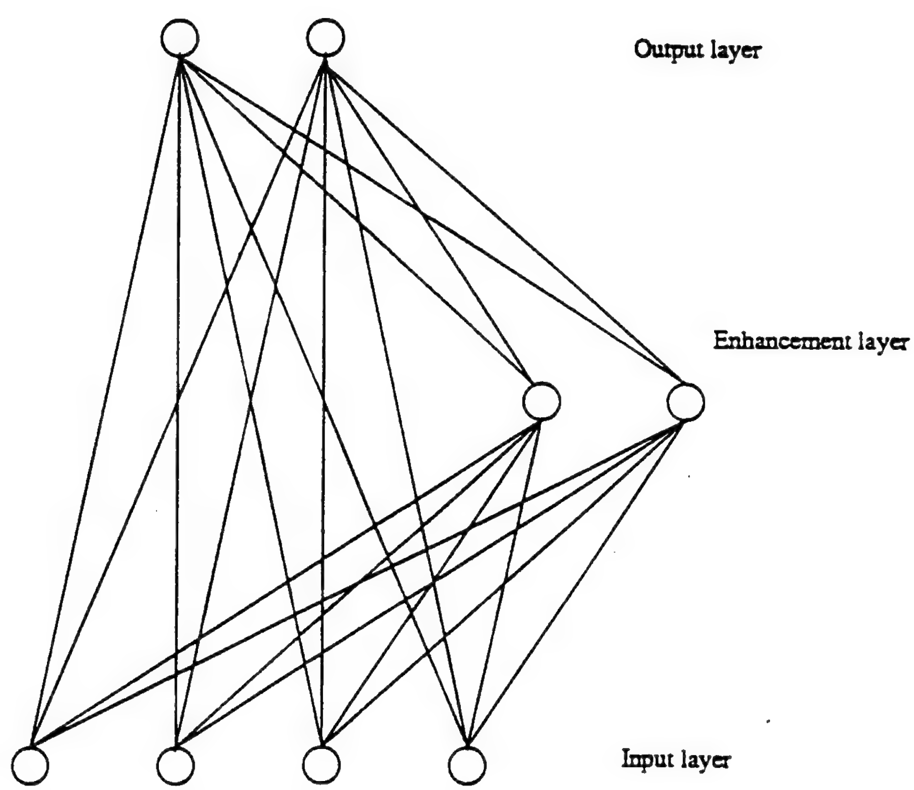
Advantages:

Rank of $[\, A \mid \sigma(AW_1)\,]$ higher than $[\, \sigma(AW_1)\,]$

More independent columns, easy to find a solution.

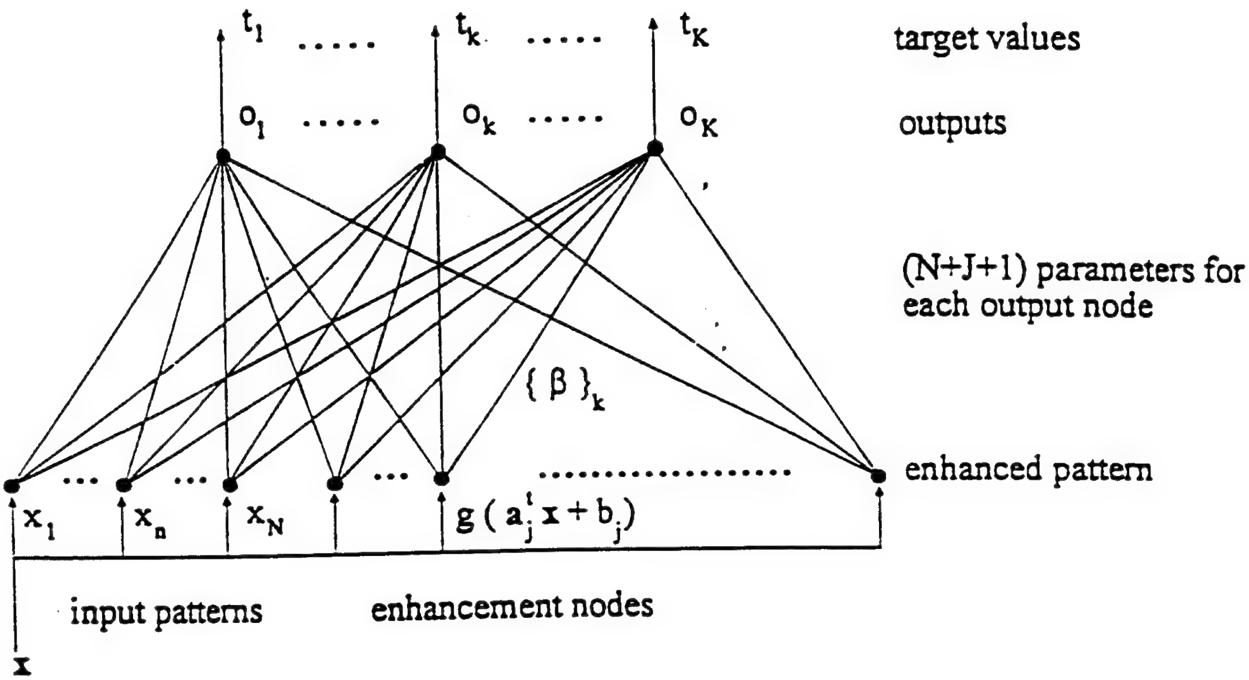
Trade-off:

Additional connections from input to output



Functional-Link Neural Network

- Architecture



- Learning Algorithm

Reiterate the Linear Algebra Theorem:

Theorem 2: Given the neural network architecture mentioned above,

(a) the network needs $q-r$ hidden nodes to have a mapping form A to B if and only if

$$rank\ ([A \mid \sigma(A\ \mathbf{W}^{bias})]) = rank\ ([\hat{A} \mid B]) = q$$

(b) the network needs maximum $N-r-1$ hidden nodes to have a unique mapping form A to B if and only if

$$rank\ ([A \mid \sigma(A\ \mathbf{W}^{bias})]) = rank\ ([\hat{A} \mid B]) = N.$$

Adding a constant bias term

- Increase the rank of the input matrix by 1

- Similar to regression equation (a constant term)

- Add even product terms to function approximation

Consider Taylor expansion of the nonlinear function:

$$\text{Taylor } (\sigma(A\,\mathbf{W}^{bias})\,) \Rightarrow \text{odd terms}$$

- If add a constant term

$$\text{Taylor } (\sigma_{ij}(w_{jo}^{bias} + \sum_{k=1}^{n} x_{ik}\, w_{jk}^{bias}))$$

, where w_{jo}^{bias} is the constant bias.

- Gives us

$$a_0 + a_1(w_{jo}^{bias} + \sum_{k=1}^{n} x_{ik}\, w_{jk}^{bias}) + a_3(\, w_{jo}^{bias} + \sum_{k=1}^{n} x_{ik}\, w_{jk}^{bias}\,)^3 + \;\cdots$$

- Further expansion on the above equation will result even product terms on x_{ik}.

- The same reason to include a bias in the σ function

$$\sigma(A\,W^{bias} + \beta)$$

Algorithm Rank-Expansion with Instant Learning (REIL).

Input: The input pattern matrix, A, the output matrix, Y, precision, and the error tolerance.

Output: The weight matrix, W, and the neural network.

Step 1. Lop off output elements, Y, within (-1,1) according to the precision.

Step 2. Calculate $B = \sigma^{-1}(Y)$.

Step 3. Add k hidden nodes and assign random weights, $k \leq N - r - 1$.

Step 4. Solve weight, W, by minimizing $||\hat{A}\mathbf{W} - B||_2$.

$$\mathbf{W} = (\hat{A}^T\hat{A})^{-1}\hat{A}^T B \quad \text{or} \quad (\mathbf{W}=\hat{A}^+B) \tag{9}$$

Step 5. If mean-squared error is not satisfied, add additional nodes and go to Step 4; otherwise, stop.

End of Algorithm REIL

2600

Iterative Learning

- Patterns are input
- Reduce computing time for $(\hat{A}^T\hat{A})^{-1}$
- Update $W(k+1)$ based on $W(k)$, given a new $a(k+1)$ and a new $b(k+1)$
- Let $\hat{A}(k+1) \triangleq [\hat{A}(k) \quad a_{k+1}]^T$
- and $B(k+1)$ be partitioned as $[B(k) \quad b_{k+1}]^T$.

$$W(k+1) = (\hat{A}(k)^T\hat{A}(k) + a_{k+1}^T a_{k+1})^{-1}(\hat{A}(k)^T B(k) + a_{k+1}^T b_{k+1}). \tag{10}$$

For convenience define:

$$P(k) = [\hat{A}(k)^T\hat{A}(k)]^{-1}, \tag{11}$$

Thus

$$W(k) = P(k)\hat{A}(k)B(k). \tag{12}$$

From matrix inversion lemma, we know

$$P(k+1) \ = [\hat{A}(k)^T\hat{A}(k) + a_{k+1}^T a_{k+1}]^{-1} \tag{13}$$

$$= P(k) - P(k)\hat{A}(k)[1 + \hat{A}(k)^T P(k)\hat{A}(k)]^{-1}\hat{A}(k)^T P(k).$$

Using Eqs. (12) and (13) in Eq. (10) yields

$$W(k+1) = W(k) - P(k)\hat{A}(k)[1 + \hat{A}(k)^T P(k)\hat{A}(k)]^{-1}$$
$$[B(k+1) - \hat{A}(k)W(k)] \tag{14}$$

For recursive evaluation of $P(k)$

- $P(k)$ is initialized by $P(0) = (\ 1/\eta)I$,

  where η is a positive real random number approaching to zero and I is an identity matrix.

- Disadvantage:

  Very difficult to find $P(k)$ if $P(k) = [\hat{A}(k)^T \hat{A}(k)]^{-1}$ is singular

  A better method is being discovered now

Robustness and Function Approximation using Weighted Least Squares

- Problem with Weighted Least Squares:

  Interpolate the training samples

- To reduce the approximation due to erroneous collection of training samples

- To eliminate outlier effect:

  An outlier datum is a datum whose functional value, y_i, is somewhat abnormal (either very big or very small) compared with rest of functional values of other points whose abscissas are in the vicinity of x_i.

  It will distort the fitted value and prevent the approximation from following the majority of the training samples.

- Approach:

  □ Define residual error: $r_i = y_i - \hat{y}_i$

  where y_i is the desired output and $\hat{y}_i$ is the estimated value.

☐ Let M be the median of the absolute values of the residuals, $M = Median\,|r_k|$

☐ The robustness weight for the training sample point $\mathbf{x}_i$ is defined to be

$$G(\mathbf{x}_k) = S\left(\frac{r_k}{\rho M}\right), \tag{18}$$

where ρ is coefficient that controls the value of the bell-shape function based on the spreadness of the residuals.

The median absolute residual, M, is a measure of how spread out the residuals are.

residual is small (compared with ρM) ⇒ robustness weight ⇒ 1

residual is large ⇒ the corresponding weight is ⇒ 0

☐ Let M be the median of the absolute values of the residuals, $M = Median\,|r_k|$

☐ The robustness weight for the training sample point $\mathbf{x}_i$ is defined to be

$$G(\mathbf{x}_k) = S\left(\frac{r_k}{\rho M} \right), \tag{18}$$

where ρ is coefficient that controls the value of the bell-shape function based on the spreadness of the residuals.

The median absolute residual, M, is a measure of how spread out the residuals are.

residual is small (compared with ρM) $\Rightarrow$ robustness weight $\Rightarrow$ 1

residual is large $\Rightarrow$ the corresponding weight is $\Rightarrow$ 0

The weighted least squares is to minimize the error

$$\phi(W, \varepsilon) = \sum_k G(\mathbf{x}_k)(\mathbf{x}_k \mathbf{w}_k^T - \mathbf{y}_k)^2, \tag{19}$$

Redefine

$\hat{\mathbf{A}} \leftarrow G^{1/2}\hat{A}$ and $Y \leftarrow G^{1/2}Y$

Similarly,

$$\mathbf{W} = (\hat{A}^T G\hat{A})^{-1}\hat{A}^T B. \tag{20}$$

The iterative solution for weighted least squares:

$$W(k+1) = (\hat{A}(k)^T G\hat{A}(k) + a_{k+1}^T a_{k+1})^{-1}(\hat{A}(k)^T GB(k) + a_{k+1}^T b_{k+1}). \tag{21}$$

and

$$P(k) = [\hat{A}(k)^T G\hat{A}(k)]^{-1}, \tag{22}$$

Simulation Results

- Error Measurement: Mean-Squared-Error (MSE)

$$MSE = \frac{1}{|\hat{A}||Y|}\sum_{A}(y_i - \hat{y}_i),$$

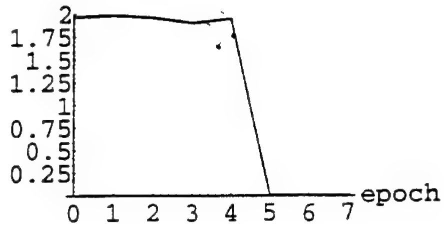
where $|\hat{A}|$ is the cardinality of $\hat{A}$, y_i is the desired output and $\hat{y}_i$ is the estimated value.

Example 1: Find the network weight for the given problem (a constant bias "-1" is added to the A matrix):

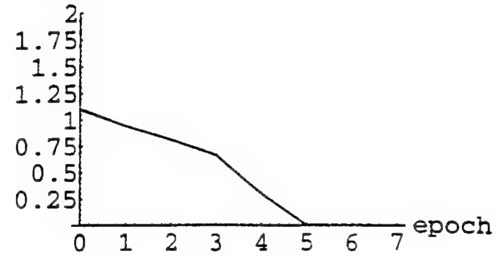
$$\hat{A} = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{3}{4} & \frac{3}{4} & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} & \frac{1}{4} & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{8} & \frac{1}{8} & \frac{1}{4} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{3}{4} & \frac{3}{4} \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix}^T,$$

$$Y = \begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \\ -1 & -1 & 1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \end{bmatrix}$$

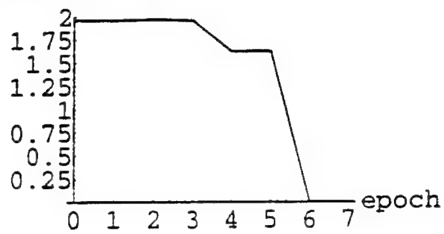
Max-error:Fig. 4



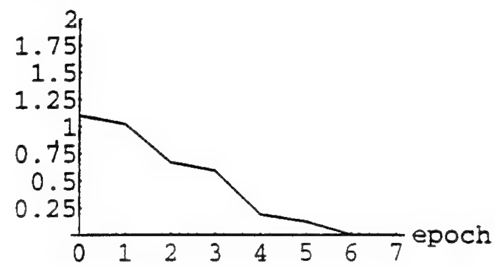
MSE:Fig. 5



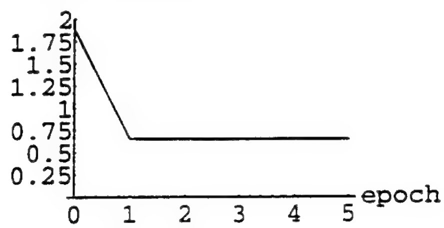
Max-error:Fig. 6



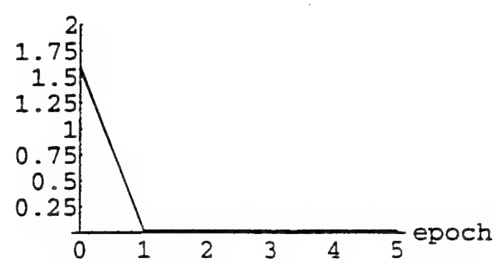
MSE:Fig. 7



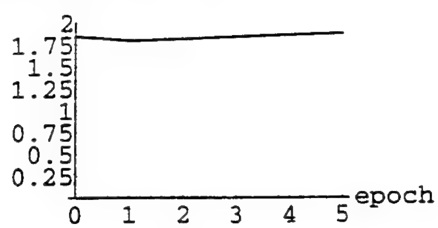
Max-error:Fig. 8



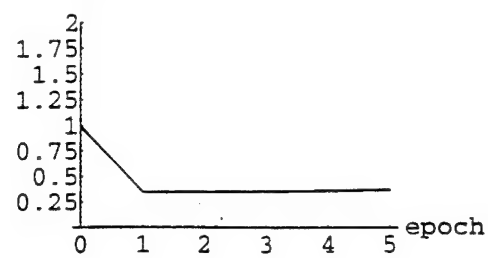
MSE:Fig. 9



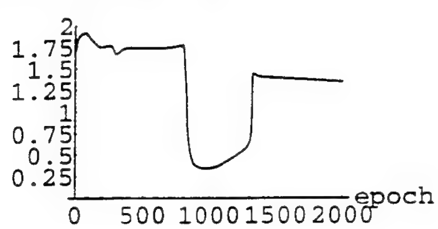
Max-error:Fig. 10



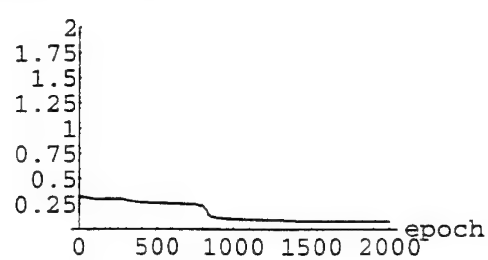
MSE:Fig. 11



Max-error:Fig. 12



MSE:Fig. 13



Example 2: Consider a nonlinear classification of a three class problem.

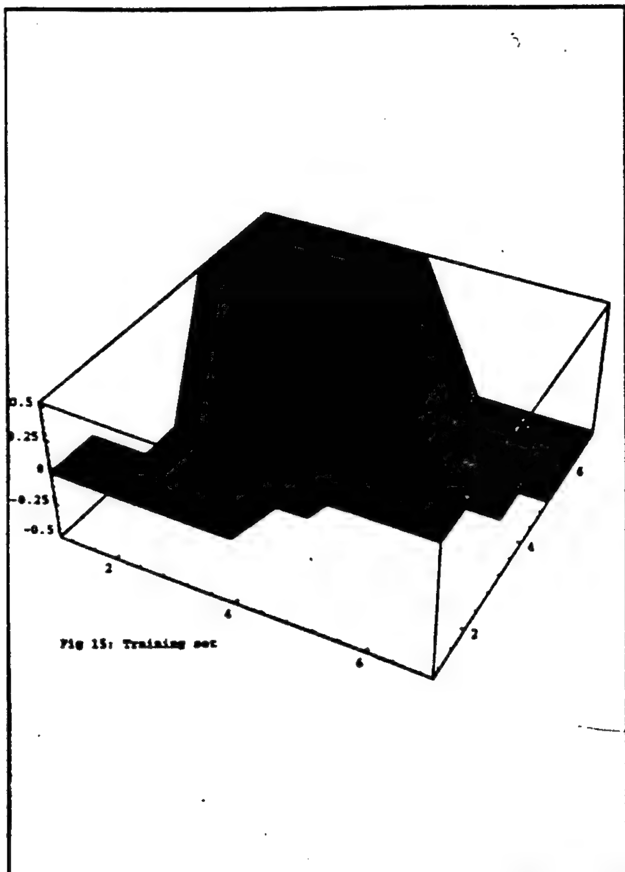


Fig 15: Training set

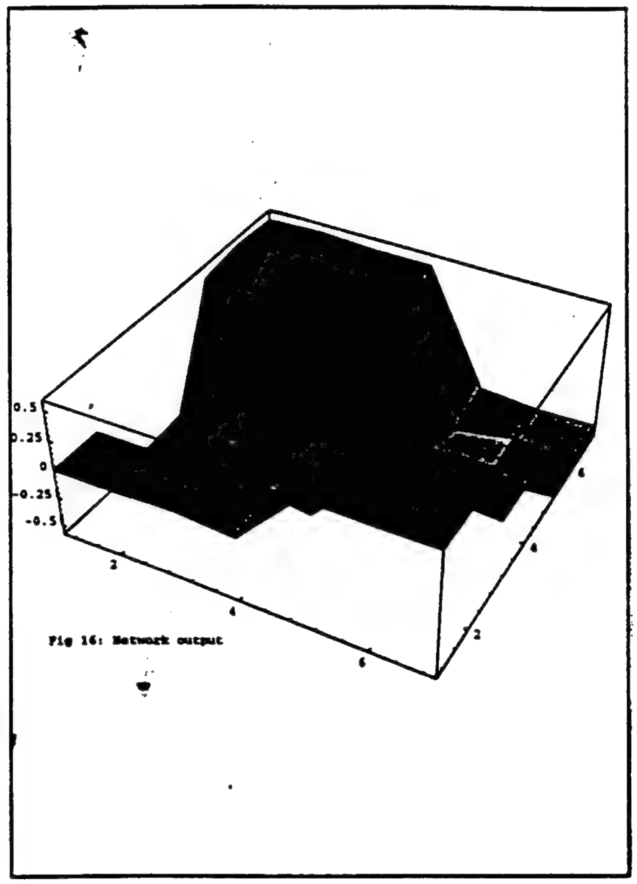


Fig 16: Network output

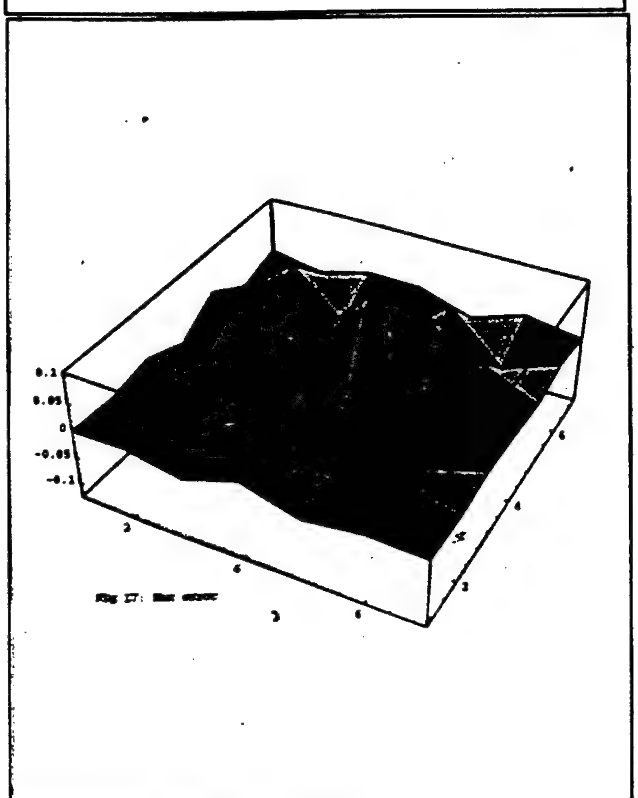
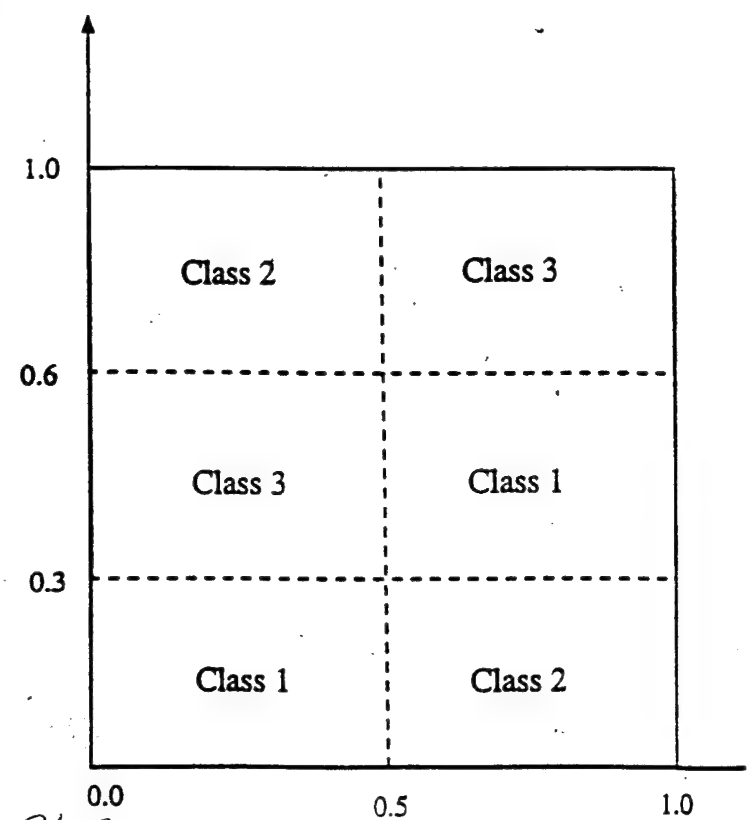


Fig 17: [illegible]



Example 3: Consider a mapping, $z = \sin(x)\sin(y)$, $x, y = 0$ to π.

☐ Training

- Forty-nine data each with 0.5 interval
- 10 hidden nodes
- Average training $\rightarrow$ 10 seconds in an Alpha machine
- The maximum error (between the desired outputs and the actual output) $\rightarrow$ around 10^{-3}.
- The mean squared error $\rightarrow$ around 10^{-5}.

☐ Testing

- 1089 data points each with 0.1 equally interval
- The maximum error (between the desired outputs and the actual output) $\rightarrow$ around 10^{-2}.
- The mean squared error $\rightarrow$ around 10^{-5}.

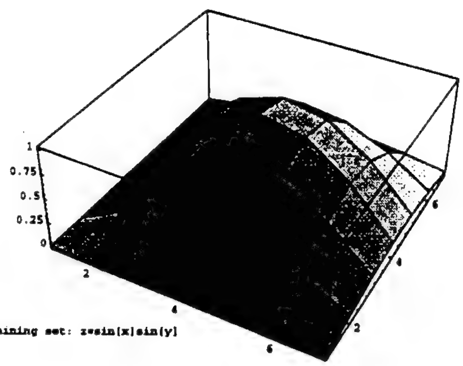


Fig. 20: Training set: z=sin[x]sin[y]

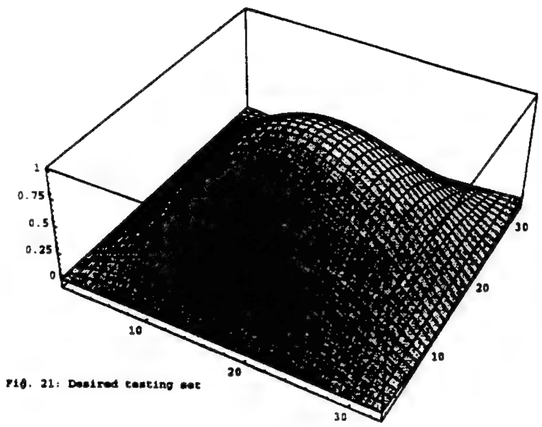


Fig. 21: Desired testing set

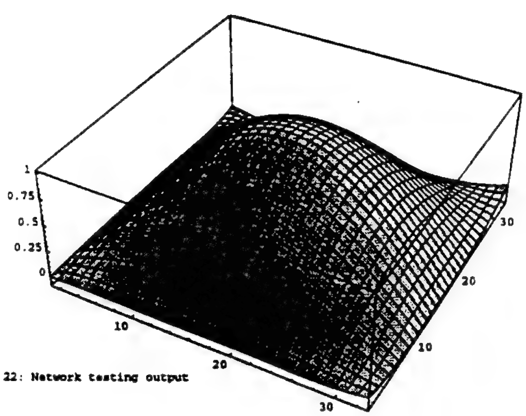


Fig. 22: Network testing output

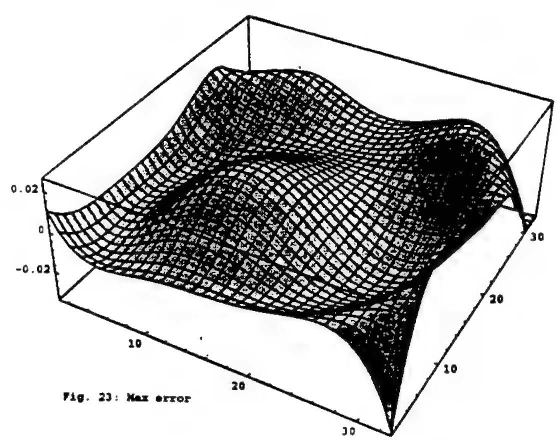


Fig. 23: Max error

Example 4: A random noise signal between [$-\mu$, μ] is added to the function in Example 3, where $\mu = 10^{-1}$.

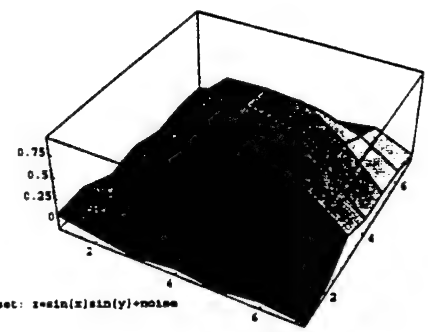


Fig. 24: Training set: z=sin(x)sin(y)+noise

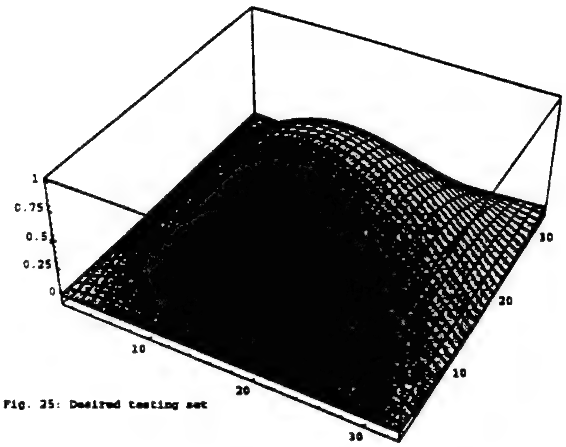


Fig. 25: Desired testing set

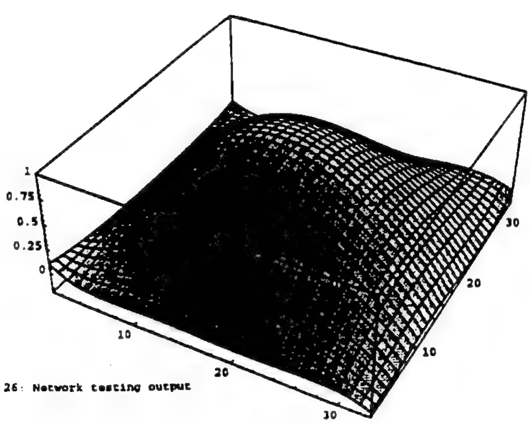


Fig. 26: Network testing output

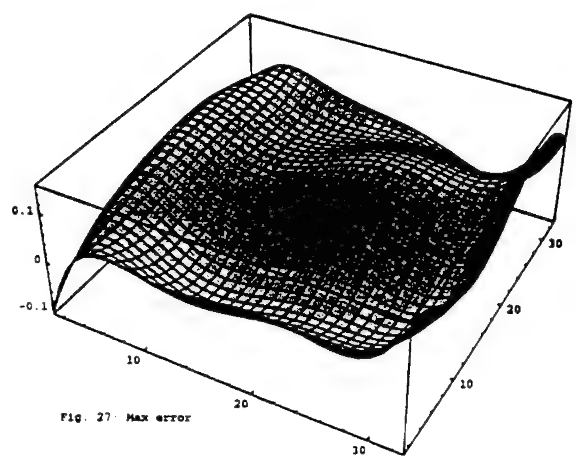
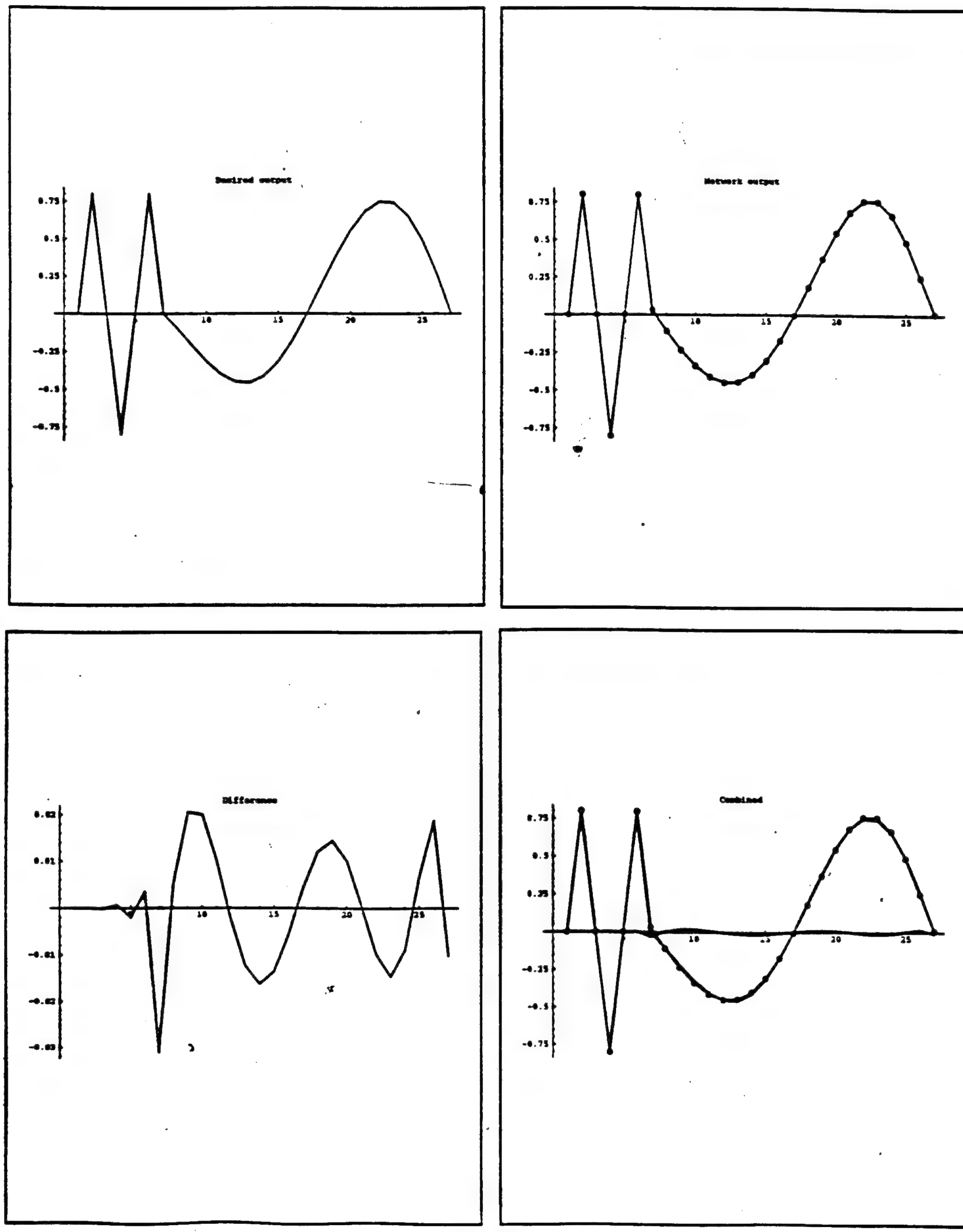
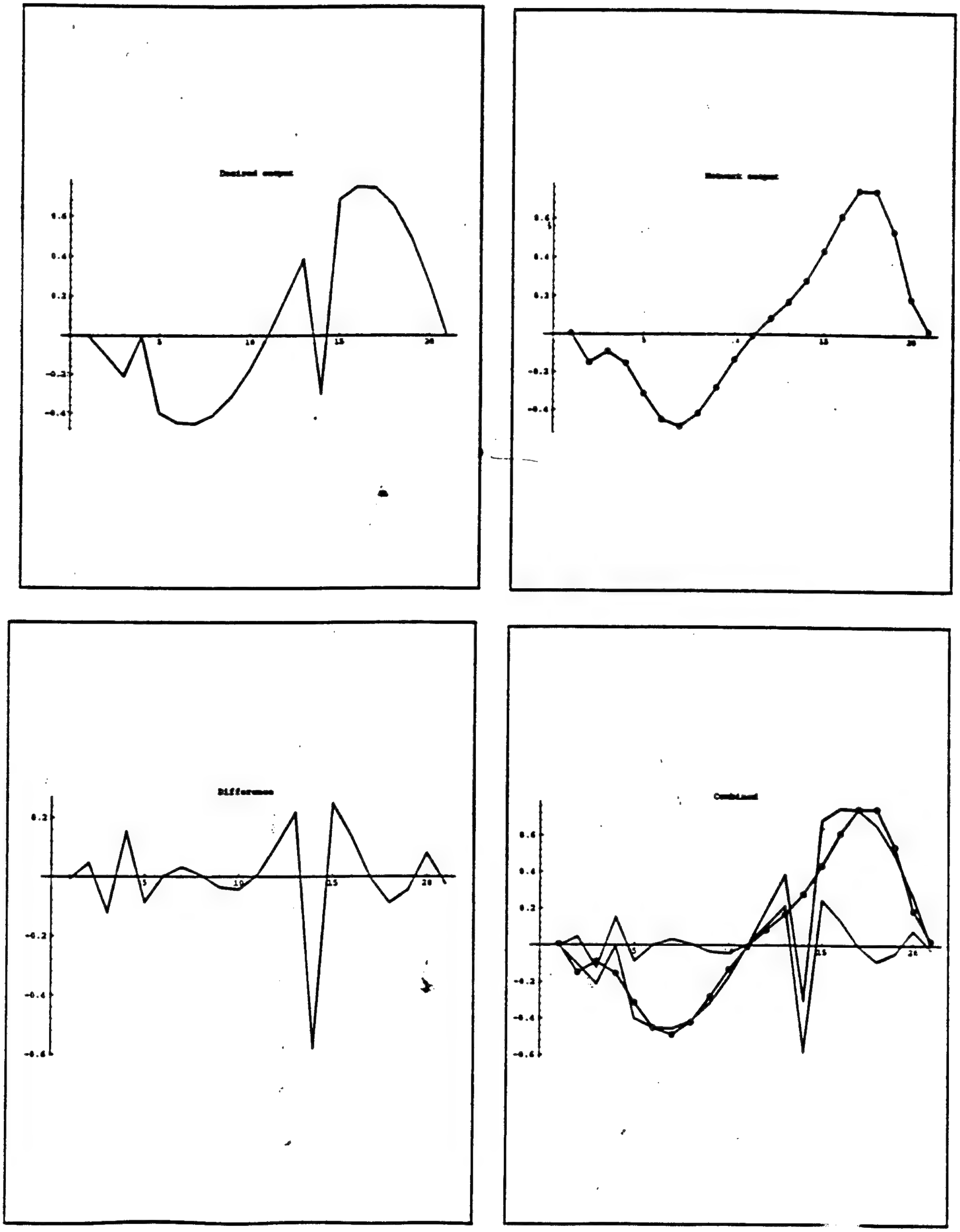


Fig. 27: Max error

Example 5: A combination of triangular and sine waves.

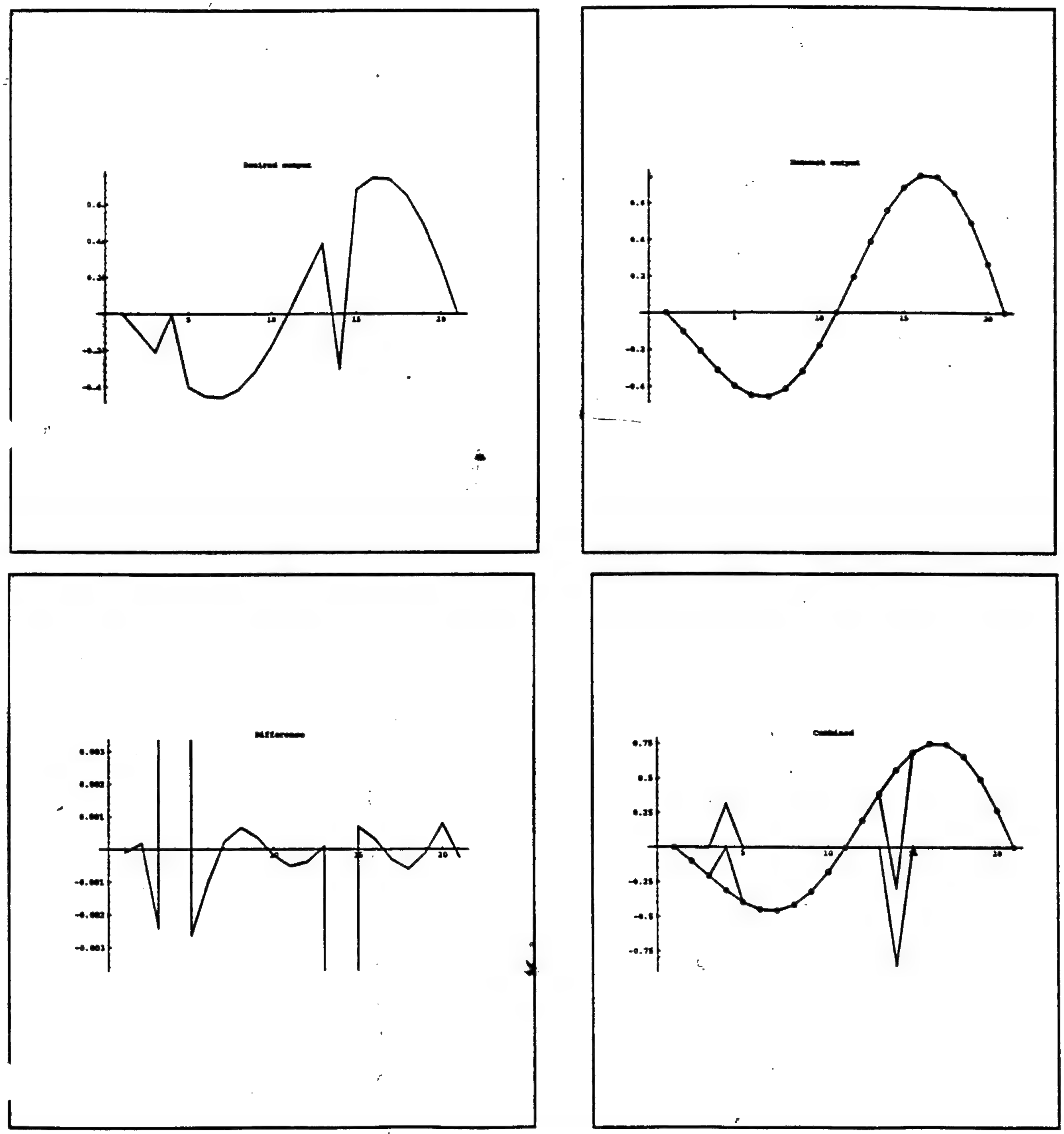


Example 6: Interpolation of a function with outliers



Example 7: Approximation of a function with outliers

- The weighted least squares method is applied
- $\rho = 6.$



Conclusion

- Rapid Learning and Iterative Learning
- Least Squares and Weighted Least Squares
    - Linear Algebra Theorem
- Function Interpolation and Approximation
- Simulation Results

Computationally Efficient Algorithms

May 24:

11:00 Constant-radius Blending of Free-Form Parametric Surfaces and New Results on Pythagorean-Hodograph Curves
R. Farouki, University of Michigan

12:00 - 1:30 Lunch

1:30 A Constraint-Driven 'Piping-Algorithm' using Cyclides
D. Dutta, University of Michigan

Constant-radius Blending of Free-Form Parametric Surfaces and New Results on Pythagorean-Hodograph Curves

Rida Farouki
University of Michigan, Ann Arbor, MI

Abstract

A key requirement in the geometrical modeling of mechanical components is the ability to smoothly "blend" two surfaces. This problem is especially difficult in the context of free-form surfaces, and has no simple solution that is compatible with the standard (Bezier/B-spline) representation schemes of CAD systems. We have developed a numerical method to construct approximate constant-radius blends between intersecting parametric surfaces that satisfy prescribed tolerances. The procedure is based on using a "pseudo-arc-length" predictor-corrector scheme to calculate the intersection of the offsets to the surfaces, and the corresponding normal projection curves on those surfaces. The blend is parameterized rationally such that its isoparametric curves are exact circular arcs that terminate with tangent continuity on the given surfaces, and are centered on the intersection of the offsets. Points on the "spine" curve of the blend are generated uniformly by arc length along it (with adaptive step reduction to accommodate for regions of high curvature) by the predictor--corrector method. This circumvents problems encountered with using increments in one of the surface parameters to trace the curve. Interpolating these points, and the points of tangency to the given surfaces, by cubic splines gives rise to a tensor--product blending surface, of degree 2 in one variable and 9 in the other, that can be represented exactly in rational Bezier/B-spline form.

We will also present recent results concerning Pythagorean-Hodograph (PH) curves, a novel class of free-form parametric curves that offer important advantages over the "ordinary" polynomial/rational curves used in CAD systems (rational offests, exactly-computable arc lengths, better curvature distributions, etc.) In particular, we describe methods to smoothly interpolate point data using PH curves, and the formulation of CNC interpolators for driving a machine tool or other device along a curvilinear path according to a feedrate specified in terms of the intrinsic geometry (curvature, arc length, etc.) of the path.

Footnotes:

1 Joint work with graduate student Ragnar Sverrisson.

Constant–radius blends for free–form parametric surfaces

Rida T. Farouki and Ragnar Sverrisson

Department of Mechanical Engineering & Applied Mechanics, University of Michigan, Ann Arbor

problem statement

- given intersecting free–form parametric surfaces $\mathbf{p}(u,v)$ and $\mathbf{q}(s,t)$
- construct constant–radius blend surface $\mathcal{S}$ that meets $\mathbf{p}(u,v)$ and $\mathbf{q}(s,t)$ with tangent continuity
- $\mathcal{S}$ is (part of) a "canal surface" — i.e., the envelope of a sphere of fixed radius d whose center traverses a space curve $\mathcal{C}$
- want to represent the blend surface $\mathcal{S}$ in rational Bézier/B–spline format for compatibility with CAD systems

geometrical considerations

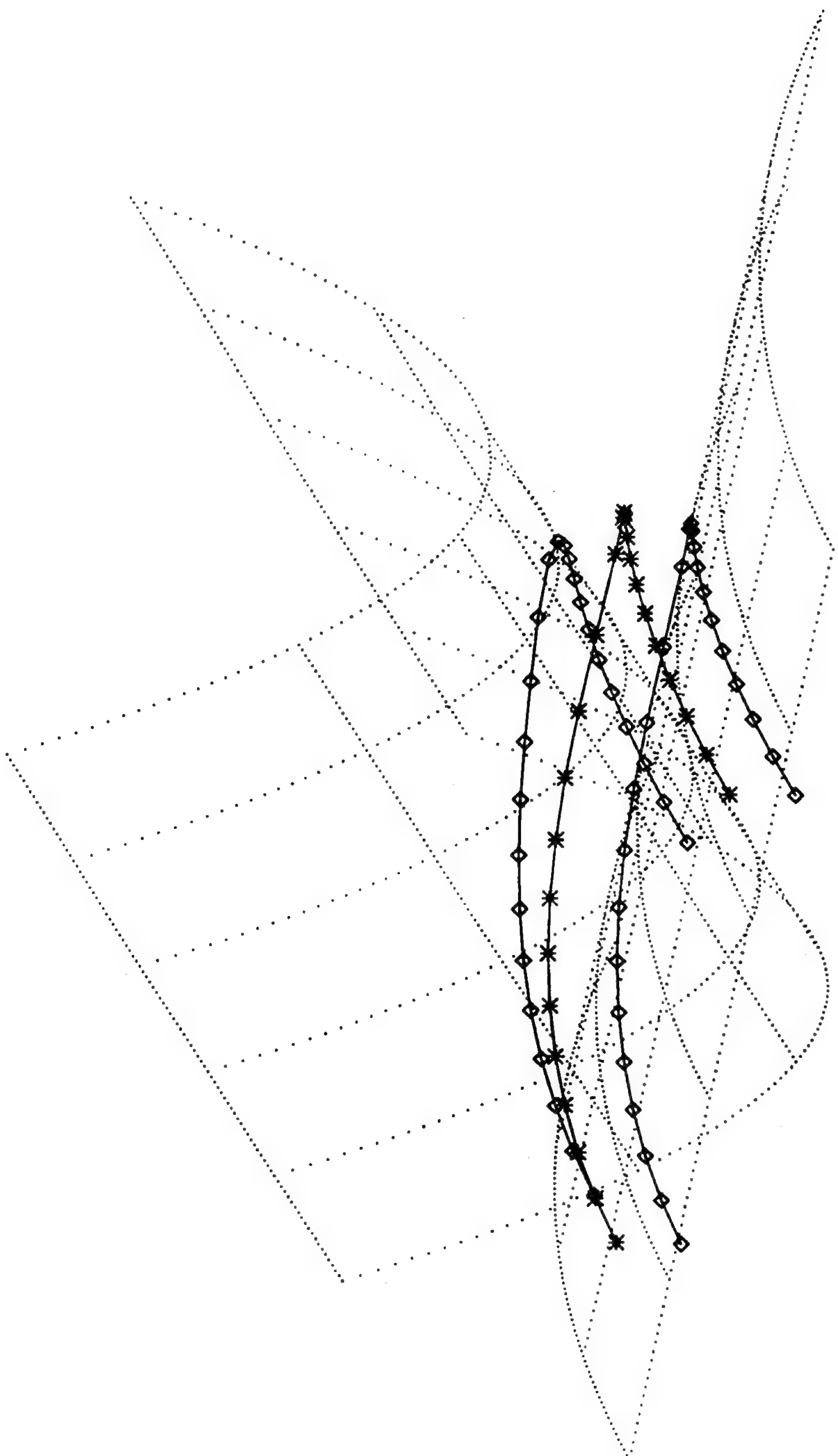
- the *spine curve* or "center–line" $\mathcal{C}$ for the blend is the intersection of the offsets

$$\begin{aligned}\mathbf{p}_d(u,v) &= \mathbf{p}(u,v) + d\,\mathbf{m}(u,v)\\ \mathbf{q}_d(s,t) &= \mathbf{q}(s,t) + d\,\mathbf{n}(s,t)\end{aligned}$$

  at distance d to the two given surfaces

- $\mathcal{C}$ represents the trajectory of the center of a ball of radius d that rolls in contact with $\mathbf{p}(u,v)$ and $\mathbf{q}(s,t)$

- the *normal projections*, $\mathcal{P}$ and $\mathcal{Q}$, of $\mathcal{C}$ onto $\mathbf{p}(u,v)$ and $\mathbf{q}(s,t)$ are "trim curves" for the blend: they indicate where it meets the given surfaces with tangent continuity



mathematical difficulties

- exact closed–form blends available only for "simple" surfaces (e.g., natural quadrics)

- want blend to be a true constant–radius canal surface of relatively low degree

- the canal surface corresponding to a general spine curve $\mathcal{C}$ is not rational (Pythagorean–hodograph space curves are an exception)

- want the trim curves $\mathcal{P}$ and $\mathcal{Q}$ to be isoparametric loci on blend, to avoid need for "trimmed surface" representations

numerical method

- "pseudo" arc–length predictor–corrector scheme gives (u, v) and (s, t) values that define sequence of points on $\mathcal{C}$ and $\mathcal{P}$, $\mathcal{Q}$

- each set of three points on $\mathcal{C}$, $\mathcal{P}$, $\mathcal{Q}$ defines a circular–arc section of the blend surface

- specially–formulated rational spline surface interpolates these sections such that all its isoparametric curves are circles of radius d centered on $\mathcal{C}$ and terminating on $\mathcal{P}$, $\mathcal{Q}$

- monitor and confine approximation errors below prescribed tolerance by an adaptive step–size procedure

offset–surface intersection

offset surfaces at distance d

$$\begin{aligned}\mathbf{p}_d(u,v) &= \mathbf{p}(u,v) + d\,\mathbf{m}(u,v)\\ \mathbf{q}_d(s,t) &= \mathbf{q}(s,t) + d\,\mathbf{n}(s,t)\end{aligned}$$

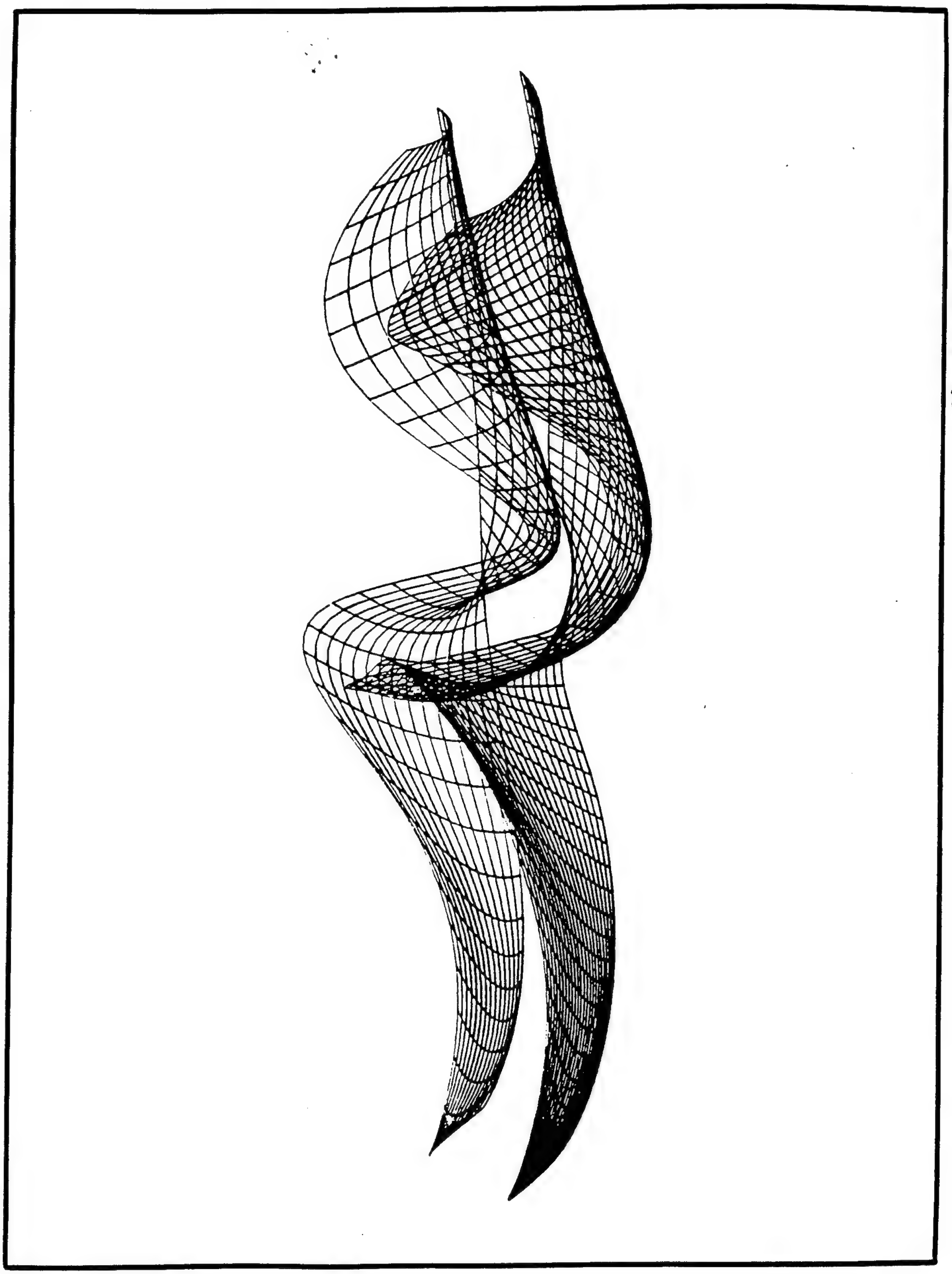
unit normal vectors are

$$\mathbf{m} = \frac{\mathbf{p}_u \times \mathbf{p}_v}{|\,\mathbf{p}_u \times \mathbf{p}_v\,|}, \quad \mathbf{n} = \frac{\mathbf{q}_s \times \mathbf{q}_t}{|\,\mathbf{q}_s \times \mathbf{q}_t\,|}$$

find values $\{u_k, v_k, s_k, t_k\}_{k=0}^{N}$ satisfying

$$\mathbf{R}_d(u,v,s,t) = \mathbf{p}_d(u,v) - \mathbf{q}_d(s,t) = \mathbf{0}$$

$$\begin{aligned}&\text{points on } \mathcal{C}: \; \mathbf{r}_k = \mathbf{p}_d(u_k,v_k) = \mathbf{q}_d(s_k,t_k)\\ &\text{points on } \mathcal{P}: \; \mathbf{p}_k = \mathbf{p}_d(u_k,v_k)\\ &\text{points on } \mathcal{Q}: \; \mathbf{q}_k = \mathbf{q}_d(s_k,t_k)\end{aligned}$$



OFFSET SURFACE

predictor step

let $\mathbf{C}(\lambda)$ be the spine curve
parameterized by its arc length λ

$u(\lambda)$, $v(\lambda)$ and $s(\lambda)$, $t(\lambda)$ define
spine curve in parameter space of surfaces

$$\mathbf{C}(\lambda) \equiv \mathbf{p}_d(u(\lambda), v(\lambda)) \equiv \mathbf{q}_d(s(\lambda), t(\lambda))$$

differentiate $\mathbf{R}_d(u, v, s, t) = \mathbf{0}$ with respect to λ

$$\frac{\mathrm{d}\mathbf{R}_d}{\mathrm{d}\lambda} = \mathbf{p}_{d,u}\frac{\mathrm{d}u}{\mathrm{d}\lambda} + \mathbf{p}_{d,v}\frac{\mathrm{d}v}{\mathrm{d}\lambda} - \mathbf{q}_{d,s}\frac{\mathrm{d}s}{\mathrm{d}\lambda} - \mathbf{q}_{d,t}\frac{\mathrm{d}t}{\mathrm{d}\lambda} = \mathbf{0}$$

$$\begin{bmatrix} \mathbf{i}\cdot\mathbf{p}_{d,v} & -\mathbf{i}\cdot\mathbf{q}_{d,s} & -\mathbf{i}\cdot\mathbf{q}_{d,t} \\ \mathbf{j}\cdot\mathbf{p}_{d,v} & -\mathbf{j}\cdot\mathbf{q}_{d,s} & -\mathbf{j}\cdot\mathbf{q}_{d,t} \\ \mathbf{k}\cdot\mathbf{p}_{d,v} & -\mathbf{k}\cdot\mathbf{q}_{d,s} & -\mathbf{k}\cdot\mathbf{q}_{d,t} \end{bmatrix} \begin{bmatrix} \frac{\mathrm{d}v}{\mathrm{d}\lambda} \\ \frac{\mathrm{d}s}{\mathrm{d}\lambda} \\ \frac{\mathrm{d}t}{\mathrm{d}\lambda} \end{bmatrix} = - \begin{bmatrix} \mathbf{i}\cdot\mathbf{p}_{d,u} \\ \mathbf{j}\cdot\mathbf{p}_{d,u} \\ \mathbf{k}\cdot\mathbf{p}_{d,u} \end{bmatrix} \frac{\mathrm{d}u}{\mathrm{d}\lambda}$$

3 equations in 4 unknowns

$$\lambda = \text{arc length} \quad \Rightarrow \quad |\,\mathrm{d}\mathbf{C}/\mathrm{d}\lambda\,| = 1$$

$$\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\lambda} = \mathbf{p}_{d,u}\,\frac{\mathrm{d}u}{\mathrm{d}\lambda} + \mathbf{p}_{d,v}\,\frac{\mathrm{d}v}{\mathrm{d}\lambda}$$

$$|\,\mathbf{p}_{d,u}\,|^2 \left(\frac{\mathrm{d}u}{\mathrm{d}\lambda}\right)^2 + 2\,\mathbf{p}_{d,u}\cdot\mathbf{p}_{d,v}\,\frac{\mathrm{d}u}{\mathrm{d}\lambda}\frac{\mathrm{d}v}{\mathrm{d}\lambda} + |\,\mathbf{p}_{d,v}\,|^2 \left(\frac{\mathrm{d}v}{\mathrm{d}\lambda}\right)^2 = 1$$

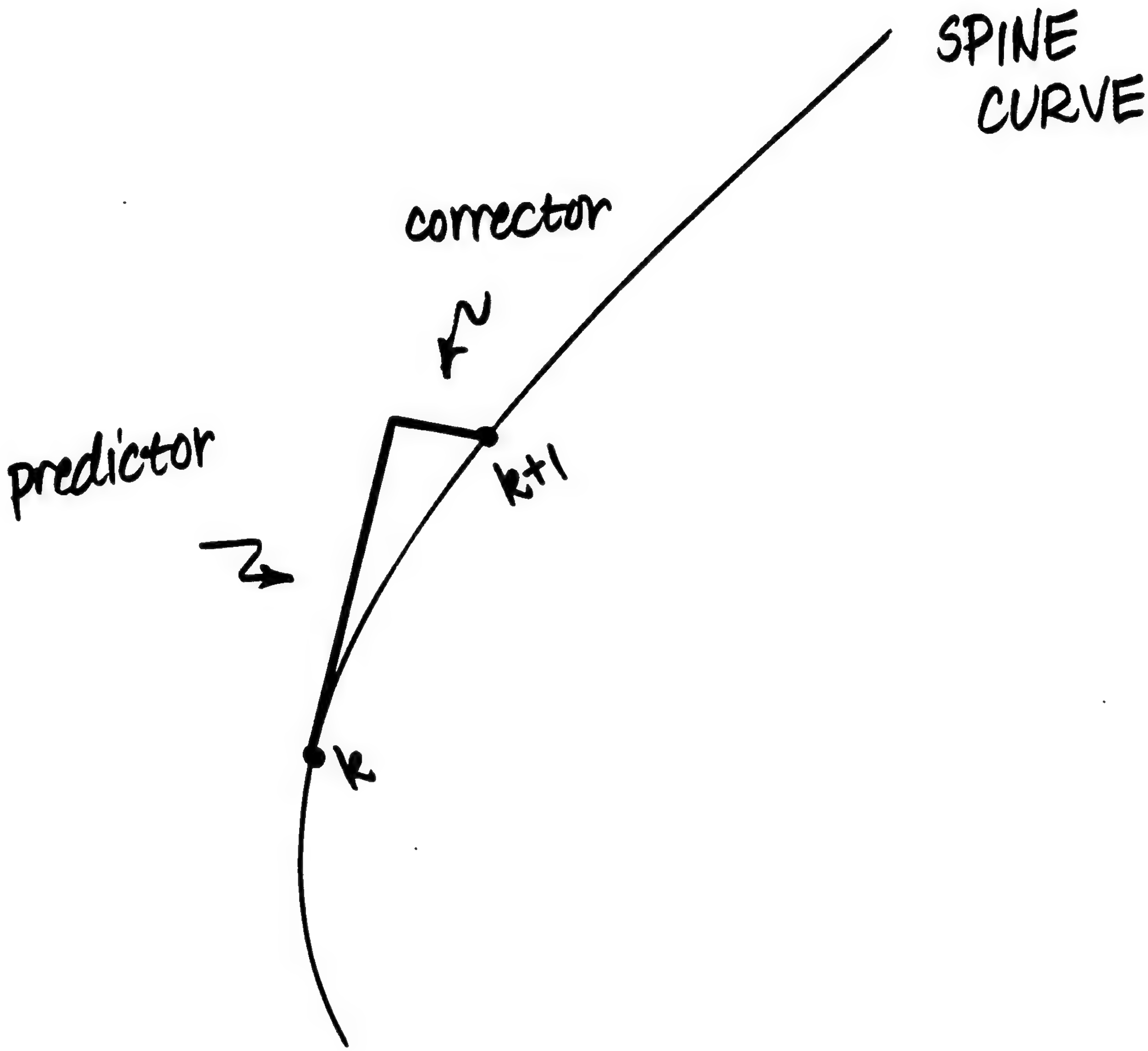
1 quadratic + 3 linear equations in 4 unknowns
$\Rightarrow$ 2 distinct solutions — choose one that gives

$$\left.\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\lambda}\right|_k \cdot \left.\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\lambda}\right|_{k-1} > 0$$

hence predictor step is

$$(u, v, s, t)^0_{k+1} = (u, v, s, t)_k + \left(\frac{\mathrm{d}u}{\mathrm{d}\lambda}, \frac{\mathrm{d}v}{\mathrm{d}\lambda}, \frac{\mathrm{d}s}{\mathrm{d}\lambda}, \frac{\mathrm{d}t}{\mathrm{d}\lambda}\right)_k \Delta\lambda$$

motion by $\Delta\lambda$ along *tangent* to $\mathbf{C}(\lambda)$


SPINE
CURVE
corrector
predictor
k+1
k

correction iterations

first–order Taylor expansion of system

$$\begin{aligned} \mathbf{R}_d(u+\Delta u, v+\Delta v, s+\Delta s, t+\Delta t) &= \mathbf{0} \\ F(u+\Delta u, v+\Delta v) &= 0 \end{aligned}$$

"pseudo" arc–length condition

$$F(u,v) \;=\; [\,\mathbf{C}(\lambda)-\mathbf{C}(\lambda_k)\,]\cdot\left.\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\lambda}\right|_k \;-\;\Delta\lambda \;=\; 0$$

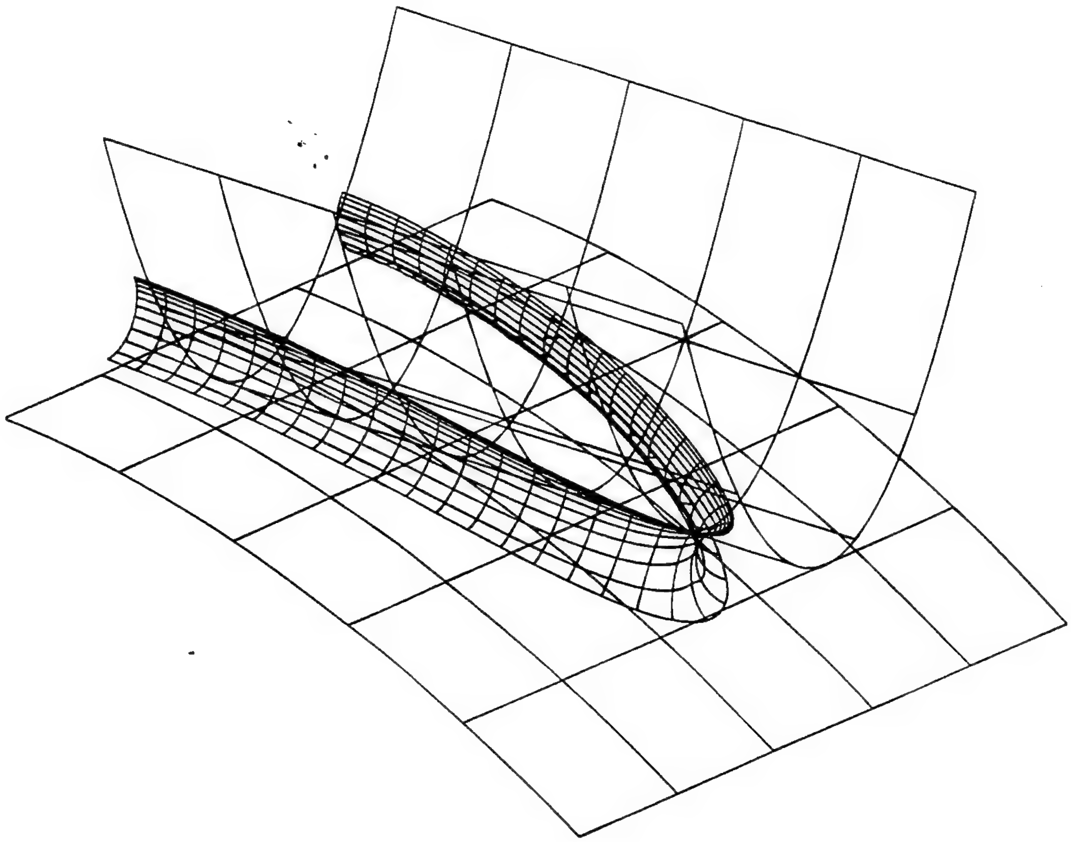
constrains motion to be *perpendicular to tangent*

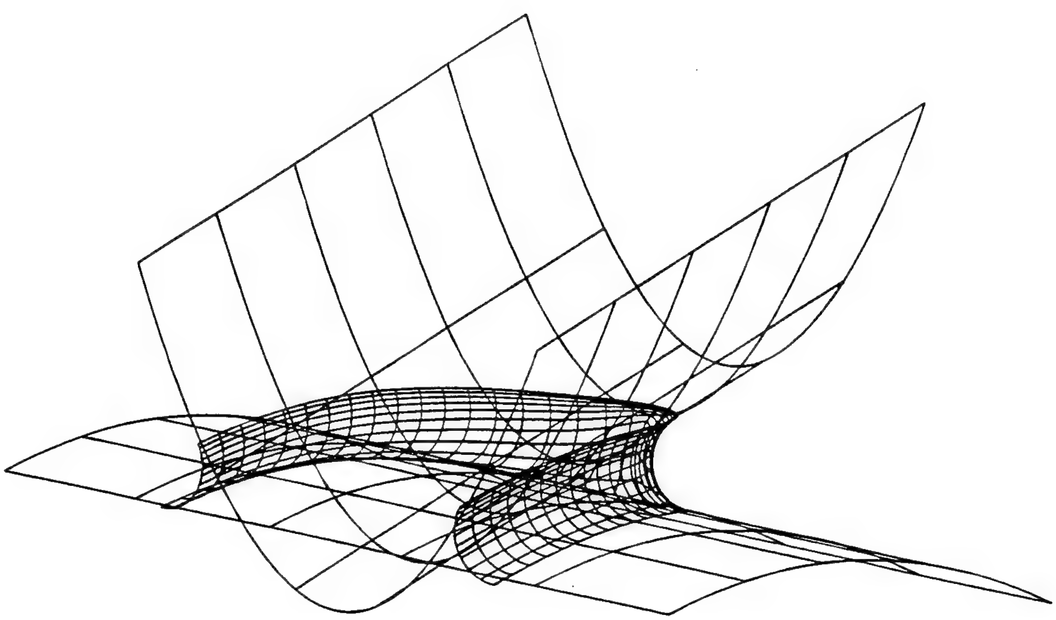
$$\begin{bmatrix} \mathbf{i}\cdot\mathbf{p}_{d,u} & \mathbf{i}\cdot\mathbf{p}_{d,v} & -\mathbf{i}\cdot\mathbf{q}_{d,s} & -\mathbf{i}\cdot\mathbf{q}_{d,t} \\ \mathbf{j}\cdot\mathbf{p}_{d,u} & \mathbf{j}\cdot\mathbf{p}_{d,v} & -\mathbf{j}\cdot\mathbf{q}_{d,s} & -\mathbf{j}\cdot\mathbf{q}_{d,t} \\ \mathbf{k}\cdot\mathbf{p}_{d,u} & \mathbf{k}\cdot\mathbf{p}_{d,v} & -\mathbf{k}\cdot\mathbf{q}_{d,s} & -\mathbf{k}\cdot\mathbf{q}_{d,t} \\ \dfrac{\mathrm{d}\mathbf{C}}{\mathrm{d}\lambda}\cdot\mathbf{p}_{d,u} & \dfrac{\mathrm{d}\mathbf{C}}{\mathrm{d}\lambda}\cdot\mathbf{p}_{d,v} & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta v \\ \Delta s \\ \Delta t \end{bmatrix} = \begin{bmatrix} -\mathbf{i}\cdot\mathbf{R}_d \\ -\mathbf{j}\cdot\mathbf{R}_d \\ -\mathbf{k}\cdot\mathbf{R}_d \\ -F \end{bmatrix}$$

solution of 4×4 system yields correction step

$$(u,v,s,t)_{k+1}^{i+1} \;=\; (u,v,s,t)_{k+1}^{i} \;+\; (\Delta u, \Delta v, \Delta s, \Delta t)$$

iterate over i until converged





blend surface construction

INPUT DATA

points $\mathbf{a}_0, \ldots, \mathbf{a}_N$ on trim curve $\mathcal{P}$
points $\mathbf{b}_0, \ldots, \mathbf{b}_N$ on trim curve $\mathcal{Q}$
points $\mathbf{c}_0, \ldots, \mathbf{c}_N$ on spine curve $\mathcal{C}$

each data set $(\mathbf{a}_k, \mathbf{b}_k, \mathbf{c}_k)$ defines circular arc of radius d from $\mathbf{a}_k$ to $\mathbf{b}_k$, centered on $\mathbf{c}_k$

want to "skin" sequence of circular arcs with *rational* canal–surface approximation

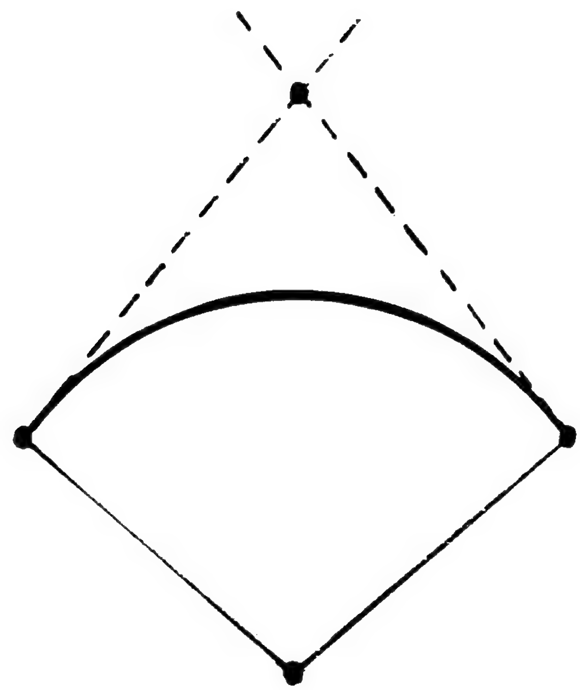
rational Bézier form of conics:

$$\mathbf{r}(u) = \frac{w_a \mathbf{a}\,(1-u)^2 + w_s \mathbf{s}\,2(1-u)u + w_b \mathbf{b}\,u^2}{w_a\,(1-u)^2 + w_s\,2(1-u)u + w_b\,u^2}$$

$\mathbf{s}$ = intersection of tangents at $\mathbf{a}$ and $\mathbf{b}$

$$\mathbf{s} = \mathbf{c} + \tfrac{1}{2}(\mathbf{a} + \mathbf{b} - 2\mathbf{c}) \sec^2 \tfrac{1}{2}\phi\,,$$

$$\text{where } \phi = \cos^{-1} \frac{(\mathbf{a} - \mathbf{c}) \cdot (\mathbf{b} - \mathbf{c})}{d^2}$$



$$\text{circular arc} \iff w_s = \sqrt{w_a w_b} \cos \tfrac{1}{2}\phi$$

a, **b**, **c** $\rightarrow$ cubic splines $\mathbf{a}(v)$, $\mathbf{b}(v)$, $\mathbf{c}(v)$
interpolating the given data

"standard" choice for weights

$$w_a = w_b = 1 \quad \text{and} \quad w_s = \cos\tfrac{1}{2}\phi$$

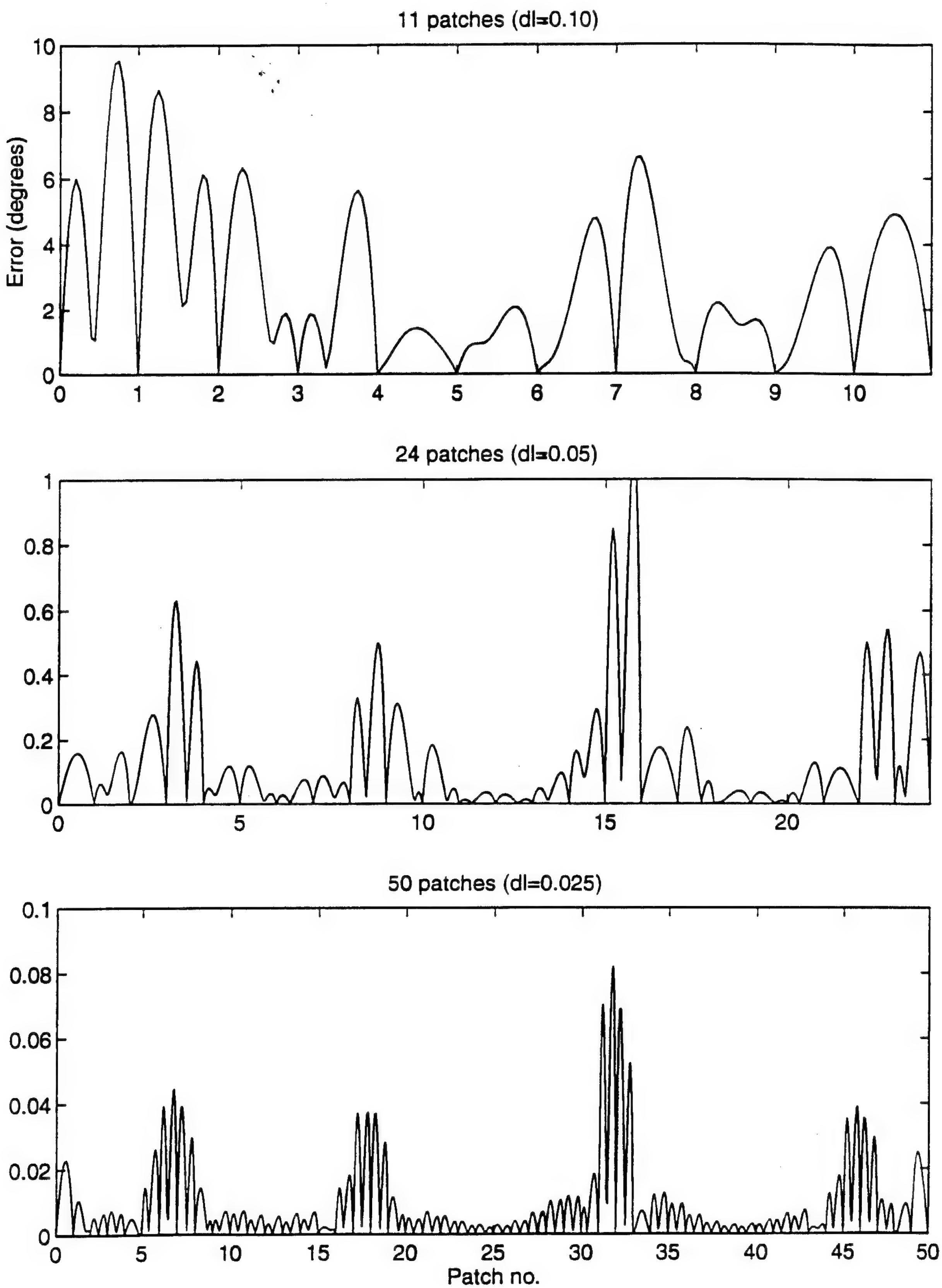
does *not* give rational surface

"non–standard" weights

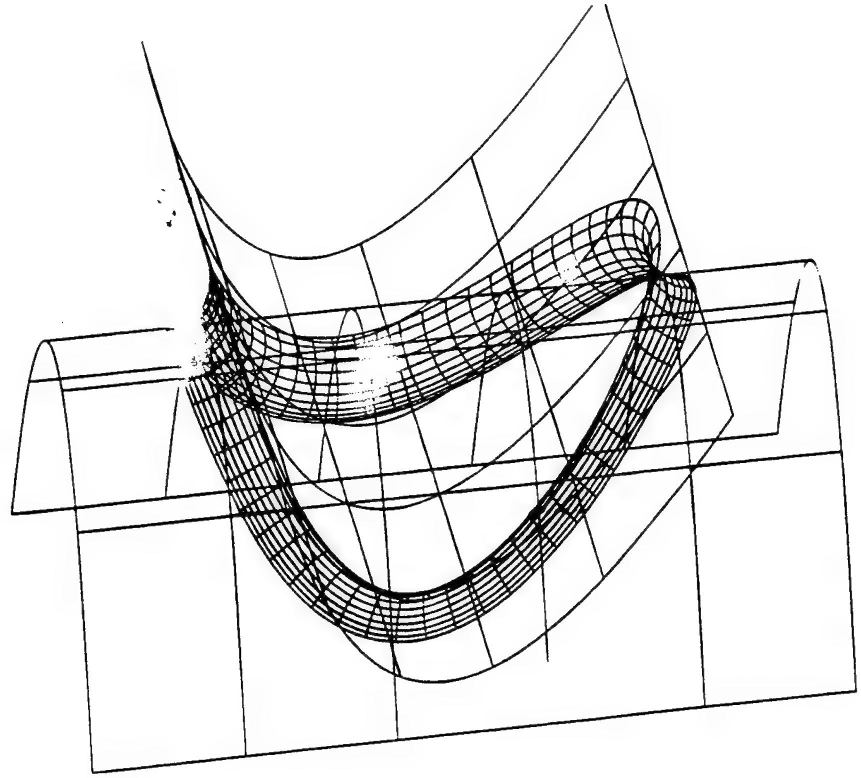
$$w_a = w_s = \cos^2\tfrac{1}{2}\phi \quad \text{and} \quad w_b = 1$$

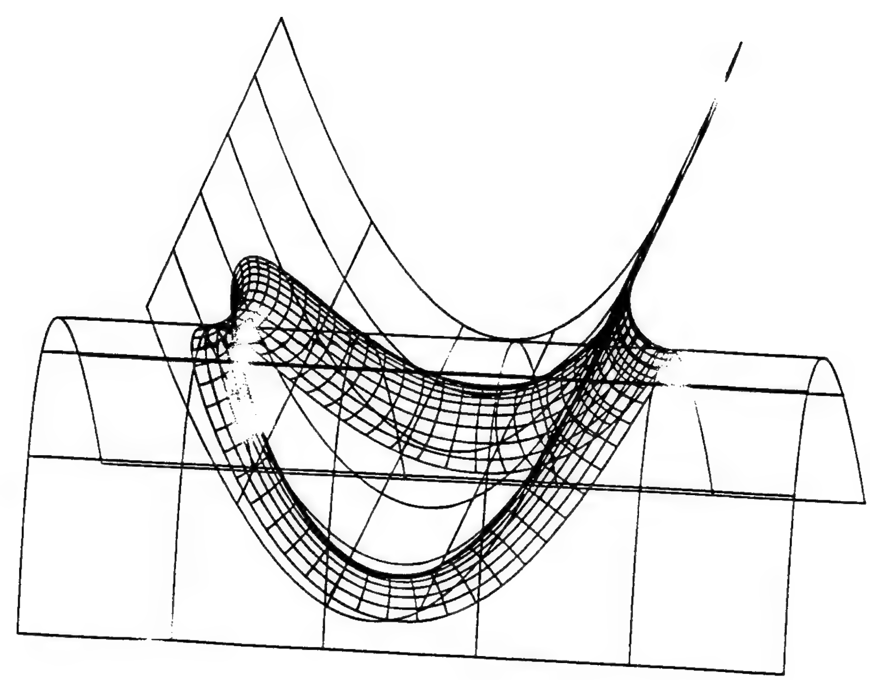
yields rational surface, degree$(u, v) = (2{,}9)$

only an "approximate" canal surface
— plane of circular iosparametric curves
not always orthogonal to spine curve $\mathbf{c}(v)$



11 patches (dl=0.10)
Error (degrees)
24 patches (dl=0.05)
50 patches (dl=0.025)
Patch no.





further work

- refine adaptive step–size selection and procedures for error control
- reliable method to select starting points for predictor–corrector scheme
- derive expressions for Bézier/B–spline control points of the blend surface
- find lower–degree representation for canal surface approximations
- exact canal surface formulations?

Parallel curves and Pythagorean hodographs

Rida T. Farouki

IBM Thomas J. Watson Research Center
Yorktown Heights, NY 10598

bibliography

Real–time CNC interpolators for Pythagorean–hodograph curves, *Comput. Aided Geom. Design*, submitted (1995).

Construction of C^2 Pythagorean–hodograph interpolating splines by the homotopy method, *Adv. Comp. Math.*, submitted (1995).

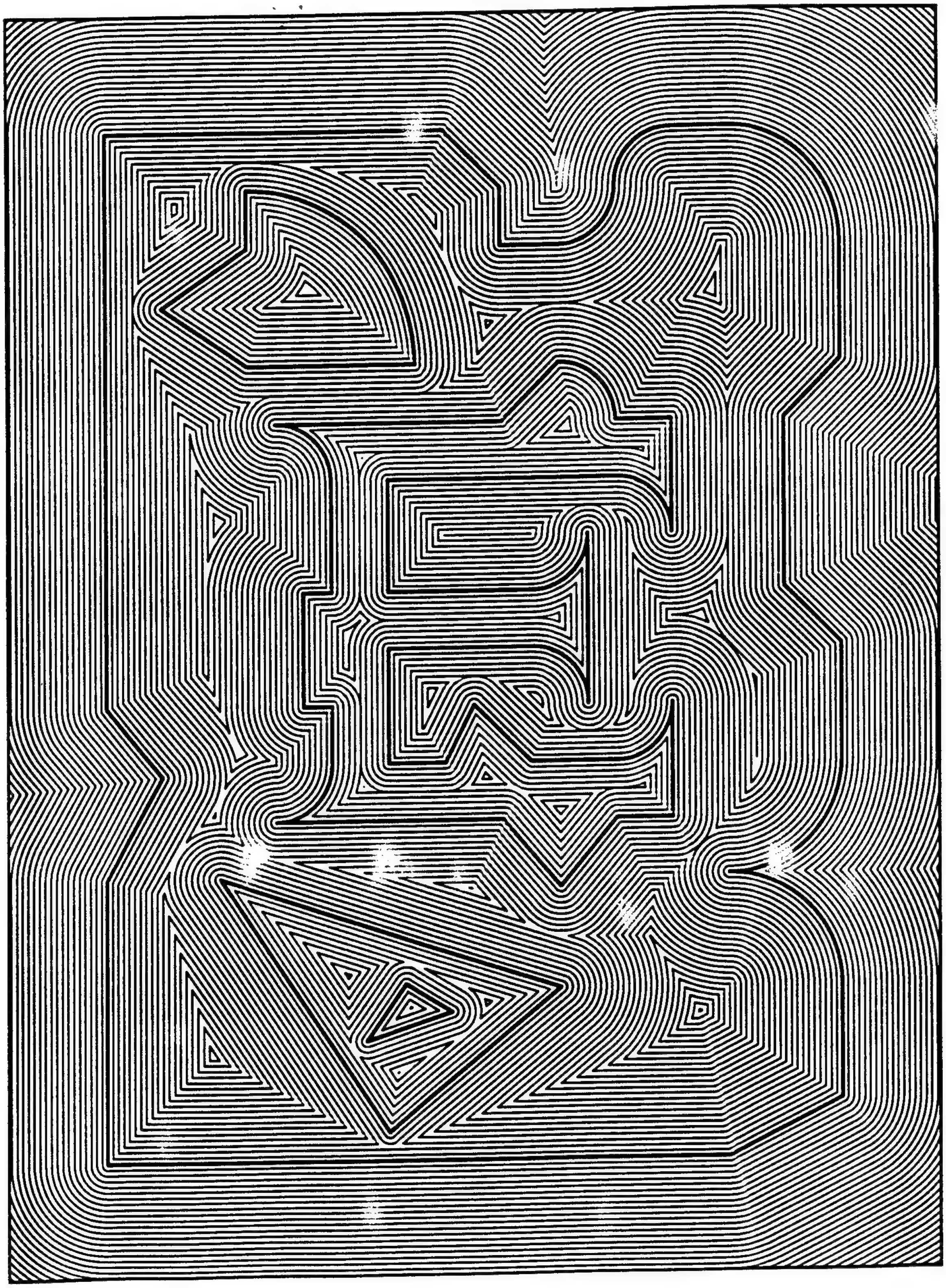
The elastic bending energy of Pythagorean–hodograph curves, *Comput. Aided Geom. Design*, to appear (1995).

Hermite interpolation by Pythagorean–hodograph quintics, *Math. Comp.*, to appear (1995).

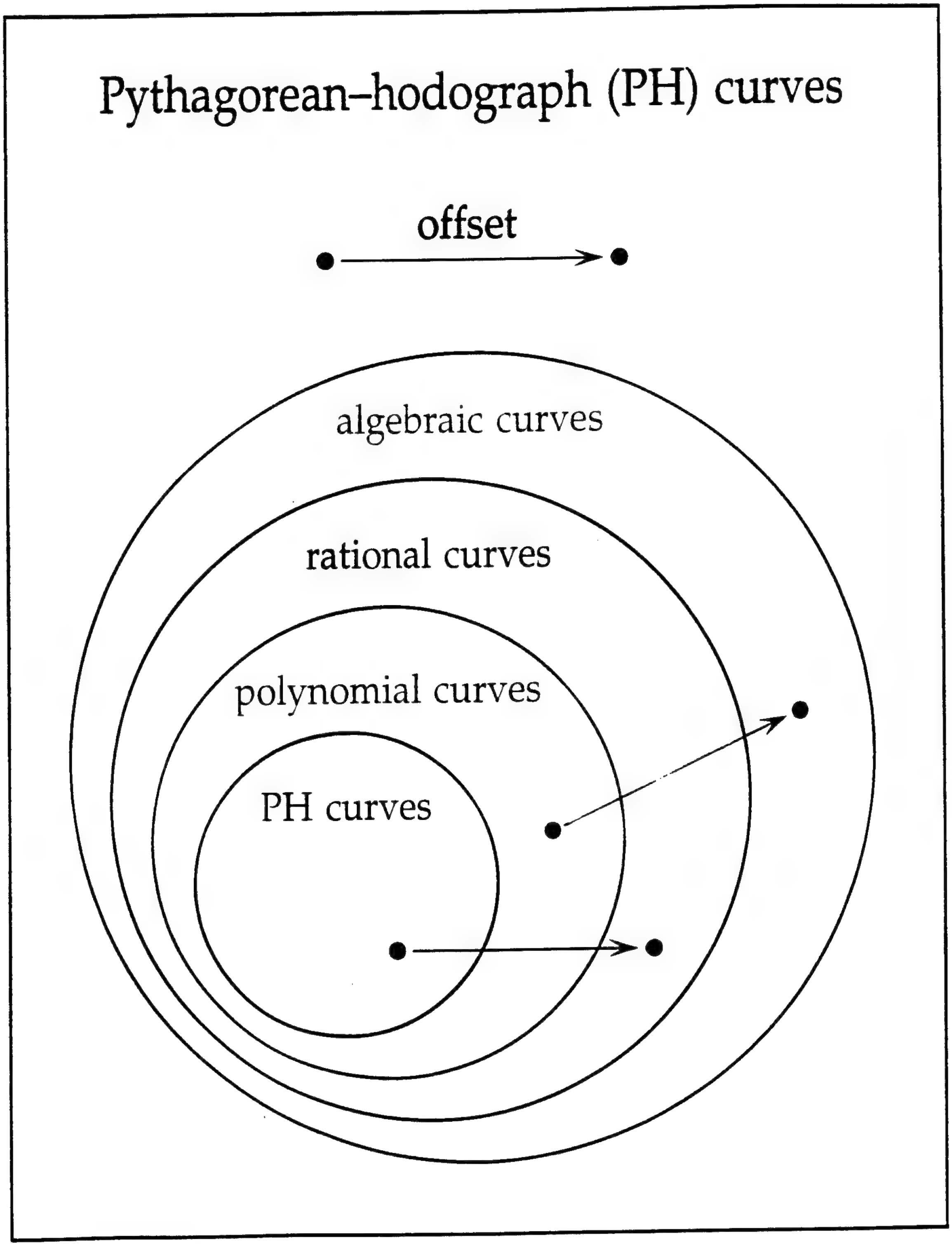
The conformal map $z \to z^2$ of the hodograph plane, *Comput. Aided Geom. Design* **11**, 363–390 (1994).

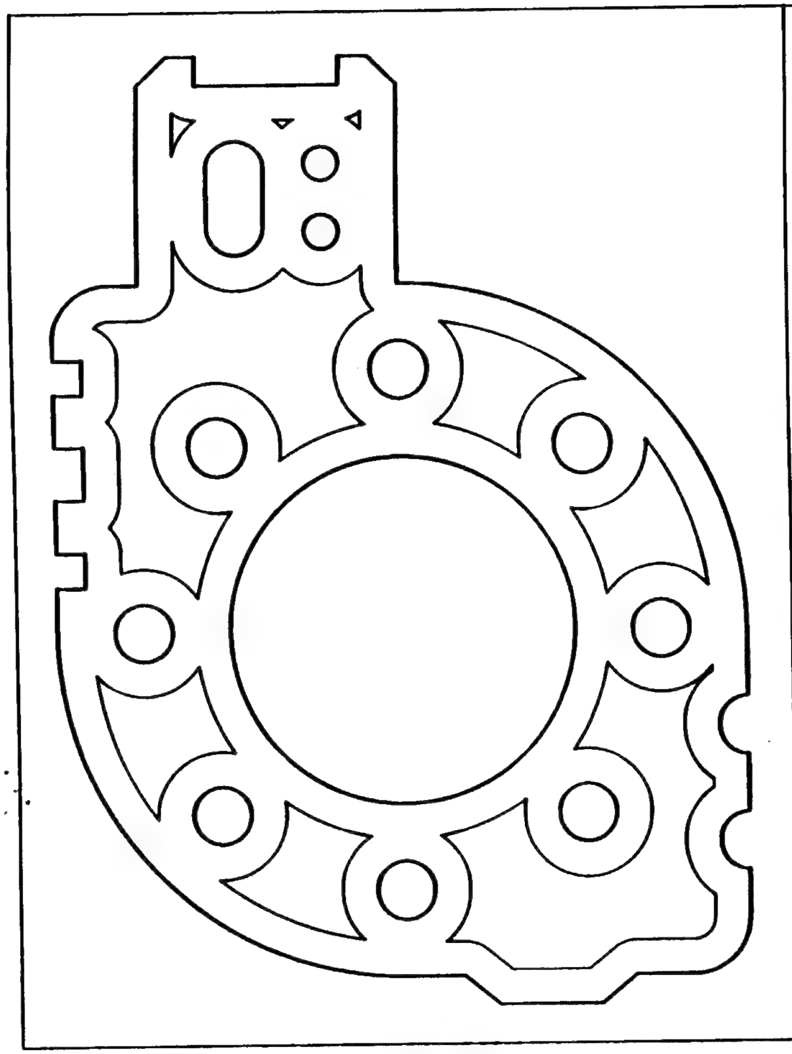
Pythagorean–hodograph curves in practical use, *Geometry Processing for Design and Manufacturing* (R. E. Barnhill, ed.), SIAM, pp. 3–33 (1992).

Pythagorean hodographs, *IBM J. Res. Dev.* **34**, 736–752 (1990).

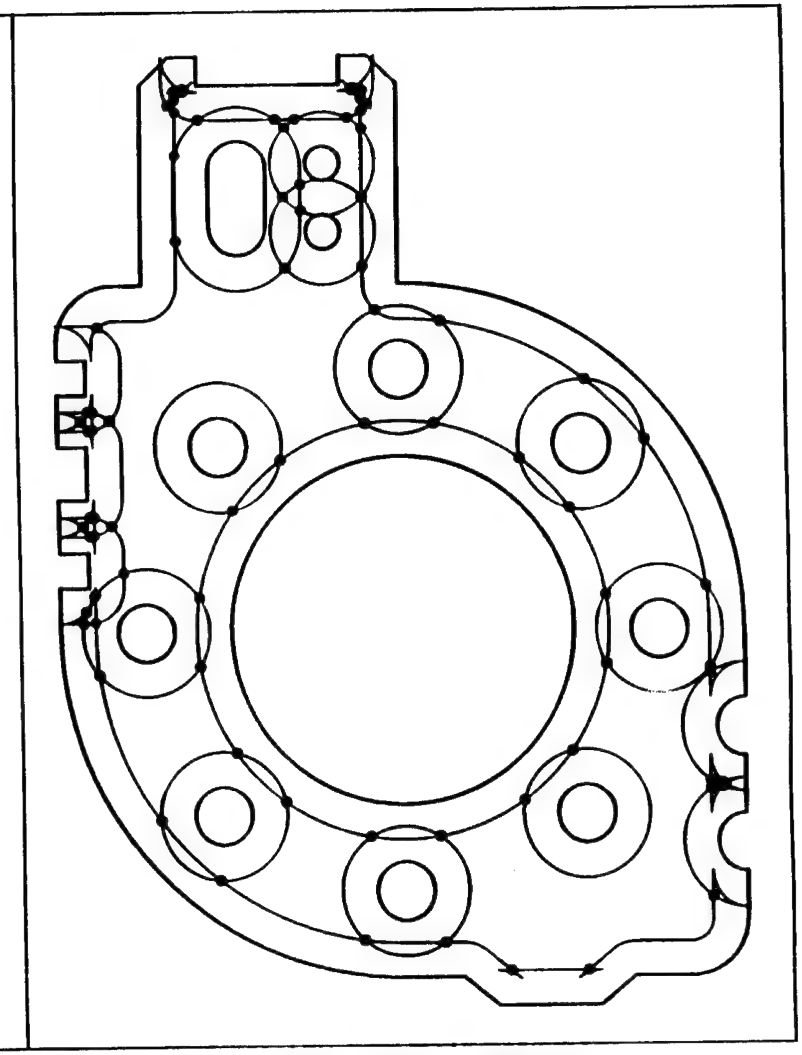


300


Pythagorean–hodograph (PH) curves
offset
algebraic curves
rational curves
polynomial curves
PH curves

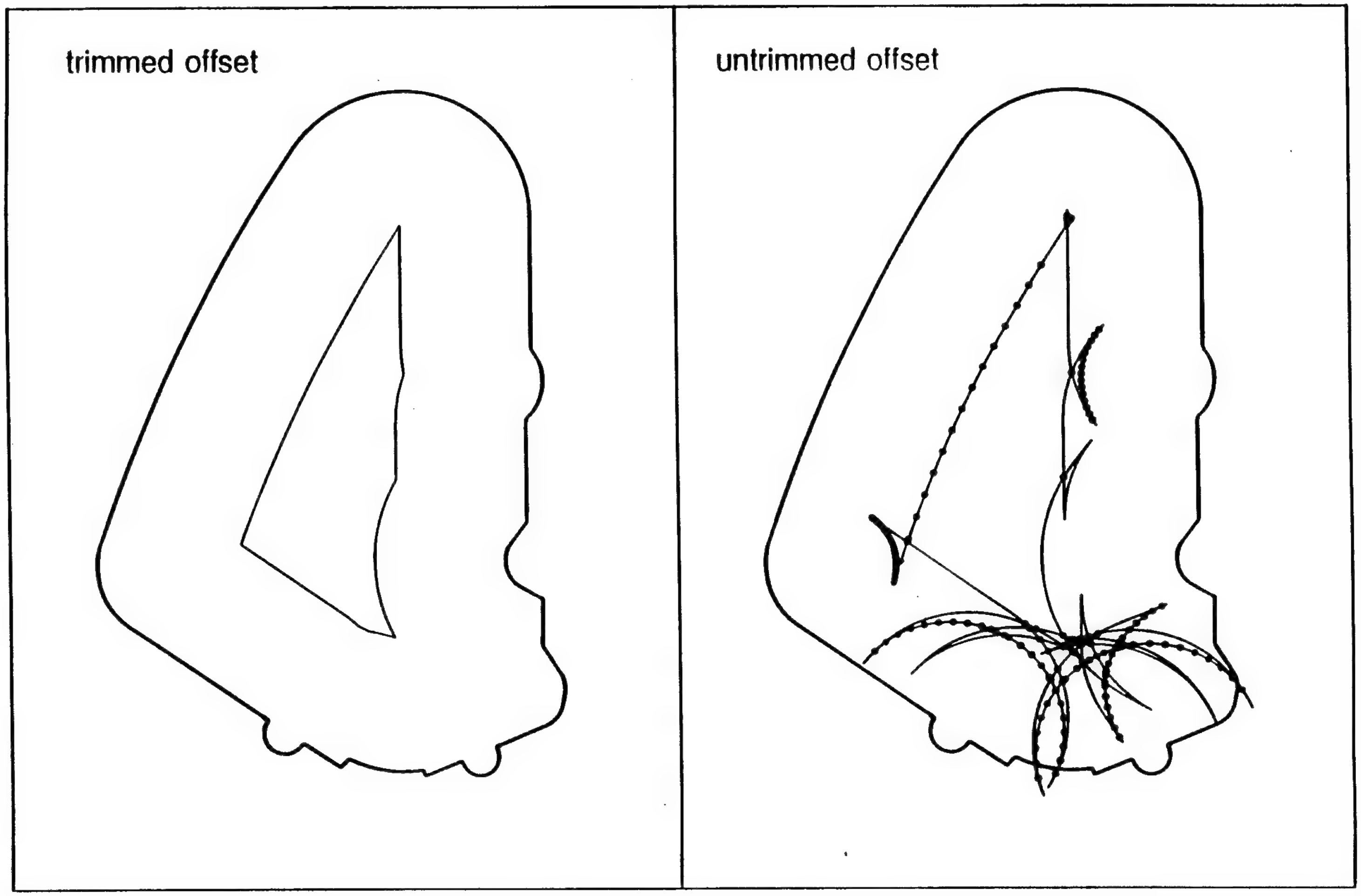


trimmed offset

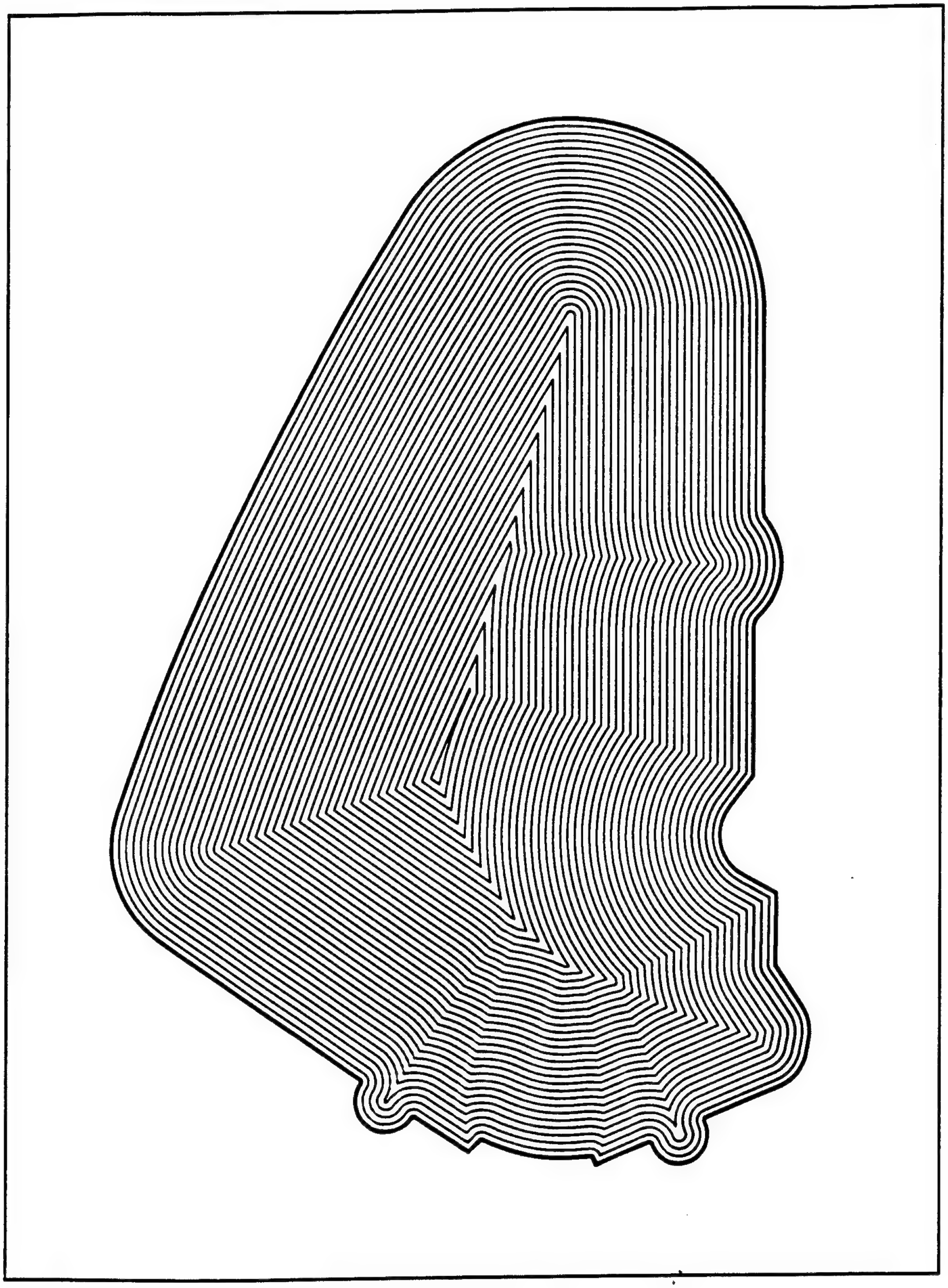


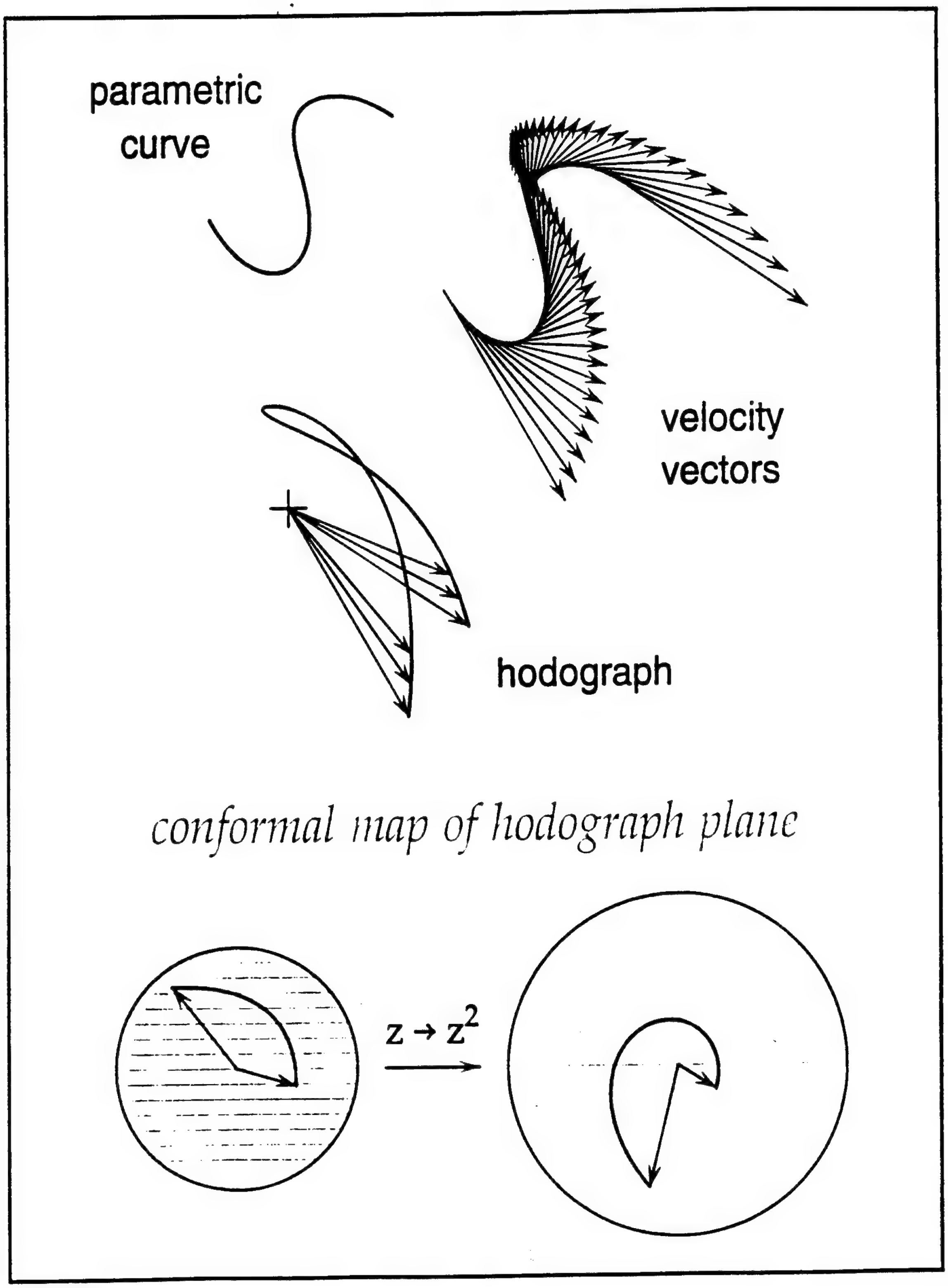
untrimmed offset

302



segments of input slice = 235
segments of untrimmed offset = 410
split segments of untrimmed offset = 10,136
segments of trimmed offset = 48




parametric curve
velocity vectors
hodograph
conformal map of hodograph plane
z → z²

polynomial hodograph

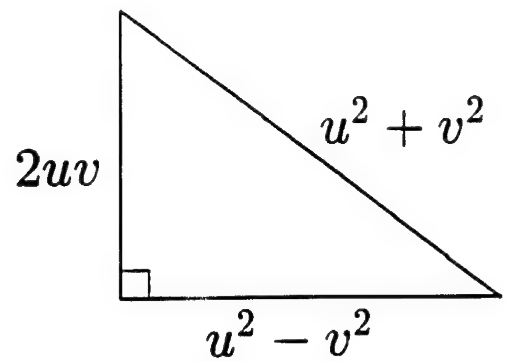
$$x'(t) = u(t)\,, \quad y'(t) = v(t)$$

maps into *Pythagorean* hodograph

$$x'(t) = u^2(t) - v^2(t)\,, \quad y'(t) = 2\,u(t)v(t)$$

integration gives PH curve $\mathbf{r}(t)$

Pythagorean polynomial triples

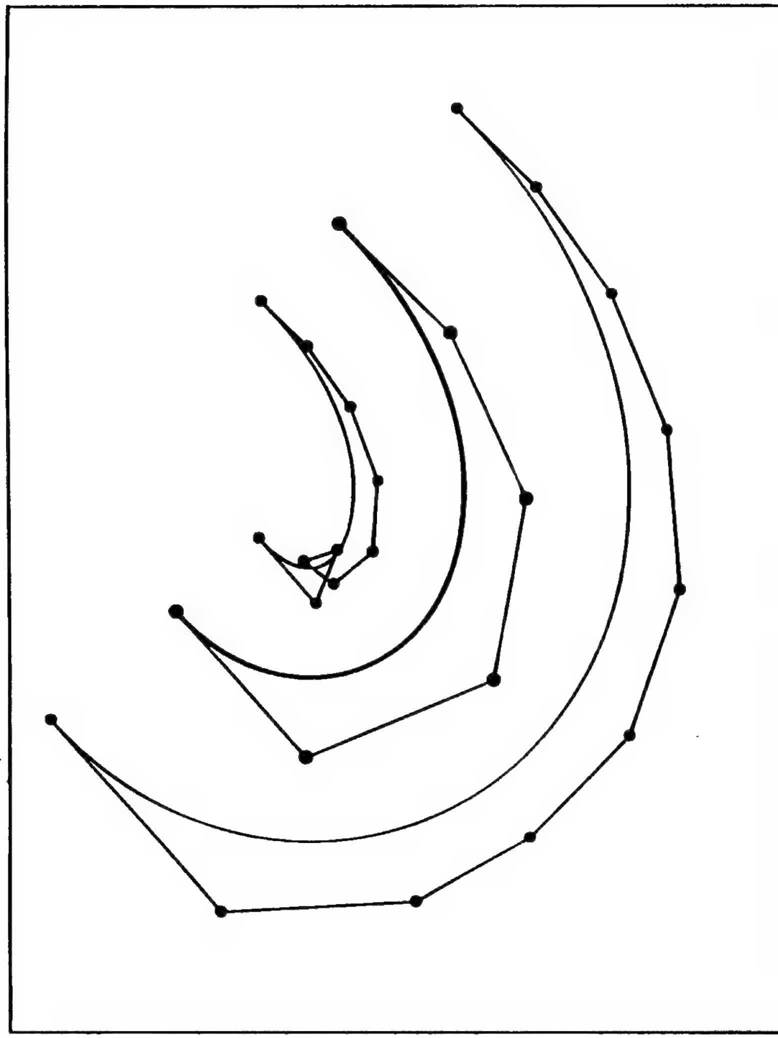


$$\mathbf{n}(t) \;=\; \frac{(y', -x')}{\sqrt{x'^2 + y'^2}} \;\rightarrow\; \frac{(2uv, v^2 - u^2)}{u^2 + v^2}$$

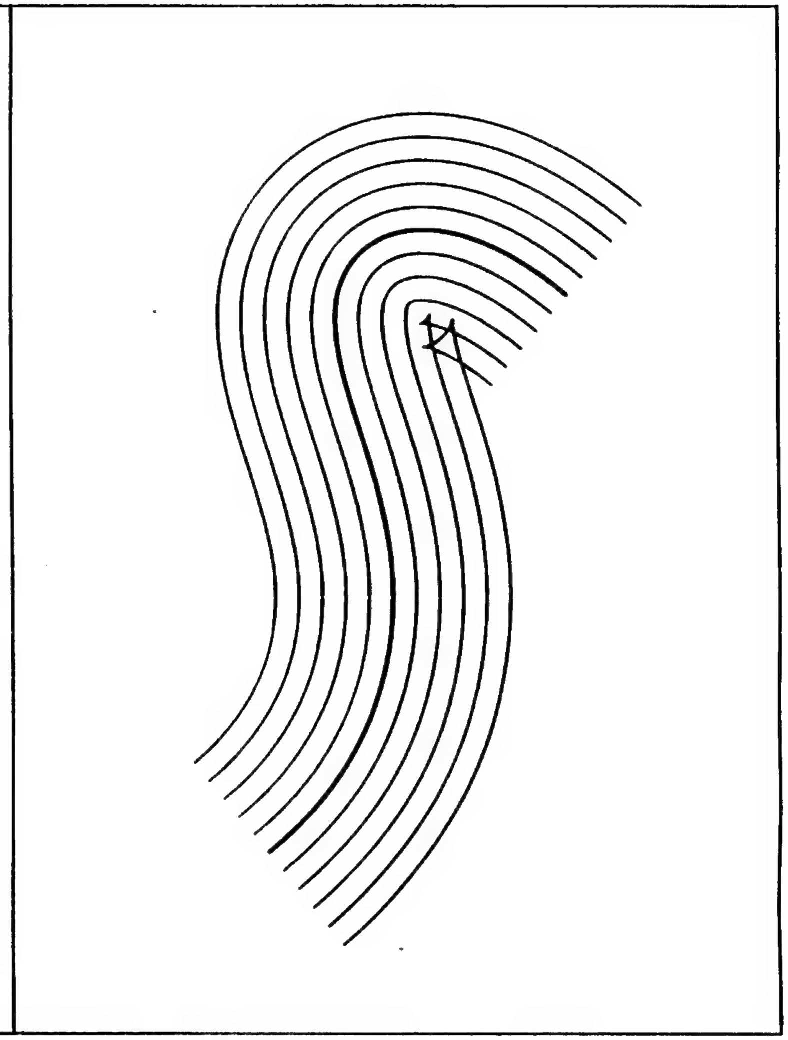
$$\boxed{\text{offset } \mathbf{r}(t) + d\,\mathbf{n}(t) \;=\; \textit{rational} \text{ curve}}$$

$$s(t) \;=\; \int \sqrt{x'^2 + y'^2}\;\mathrm{dt} \;\rightarrow\; \int u^2 + v^2\;\mathrm{dt}$$

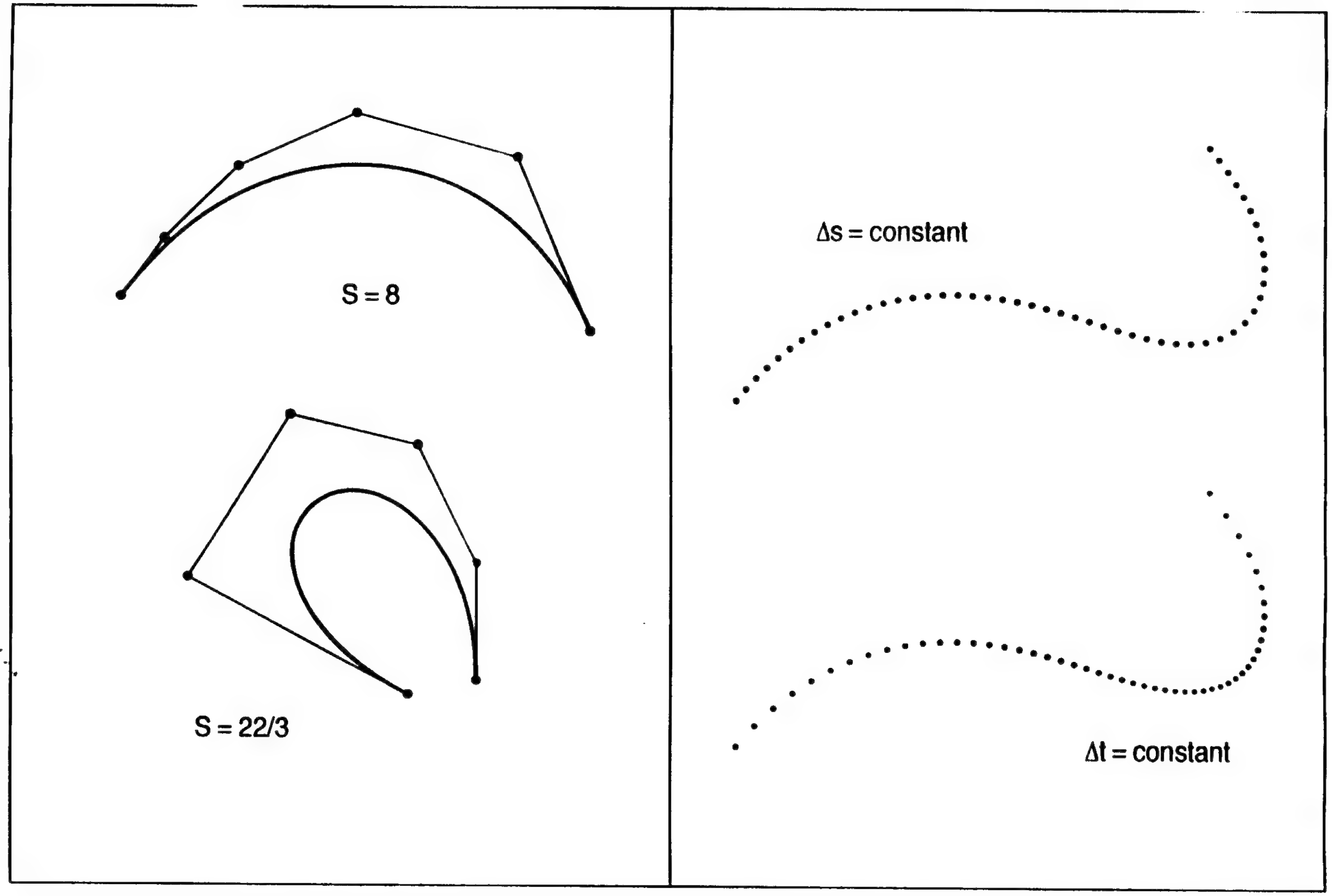
$$\boxed{\text{arc length} \;=\; \textit{polynomial} \text{ function of } t}$$



Bezier control polygons of rational offsets



offsets exact at any distance



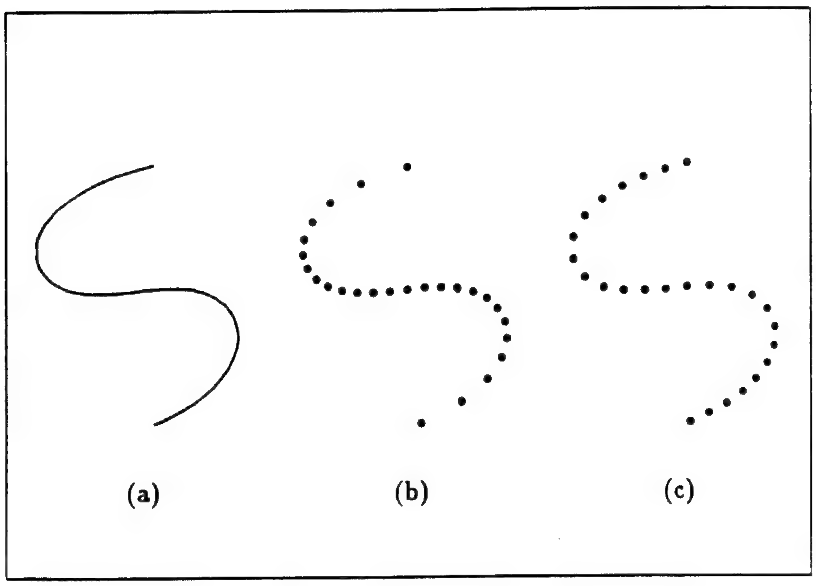
exact arc lengths

uniform arc–length rendering

308

real–time CNC interpolators for Pythagorean–hodograph curves

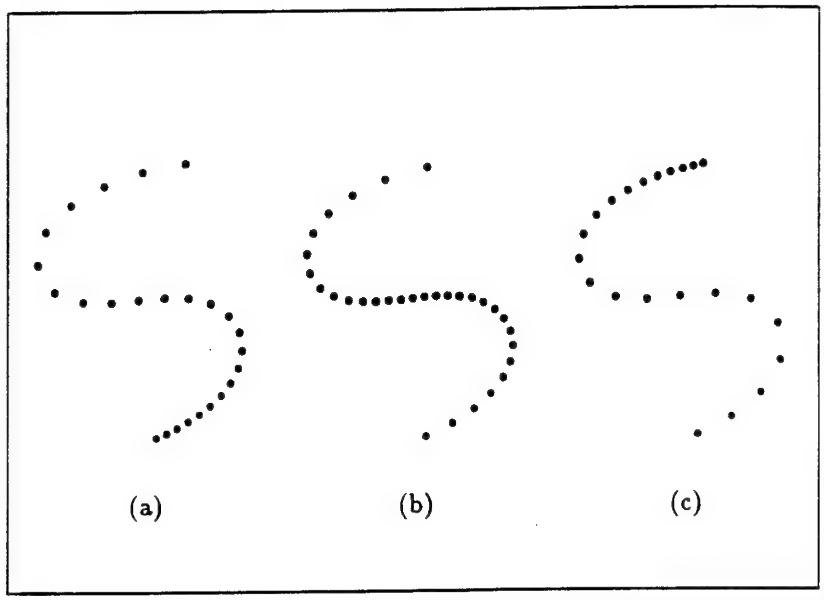
specify tool feedrate V along PH curve
as function of arc length s or curvature κ



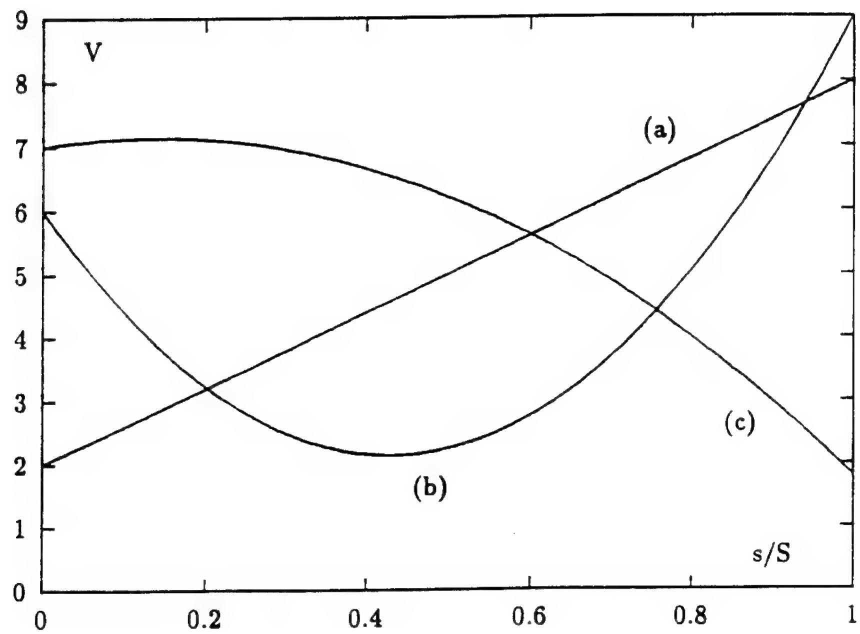
CNC "interpolator" generates reference points along curve at sampling interval Δt according to specified feedrate V

analytically reducible to sequence of (monotone) polynomial equations in the following cases —

V inversely proportional to κ
V constant, linear, or quadratic in s

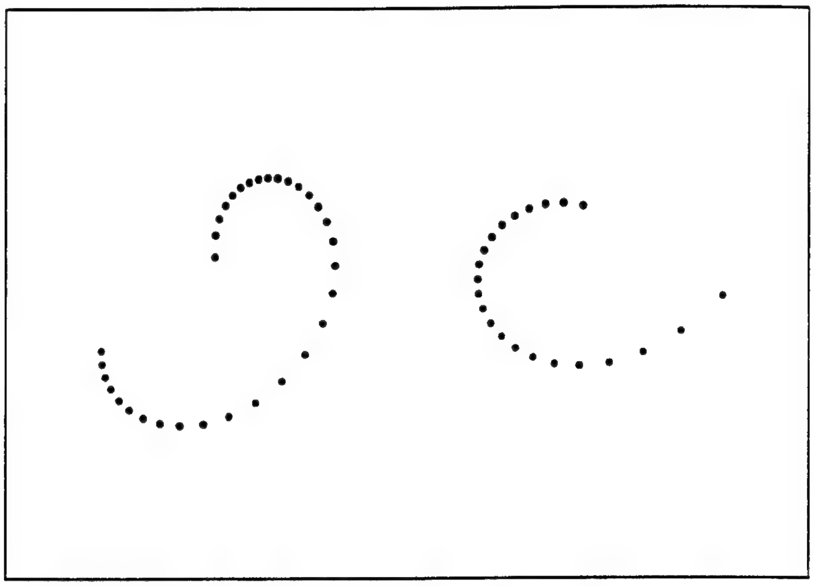


(a)
(b)
(c)

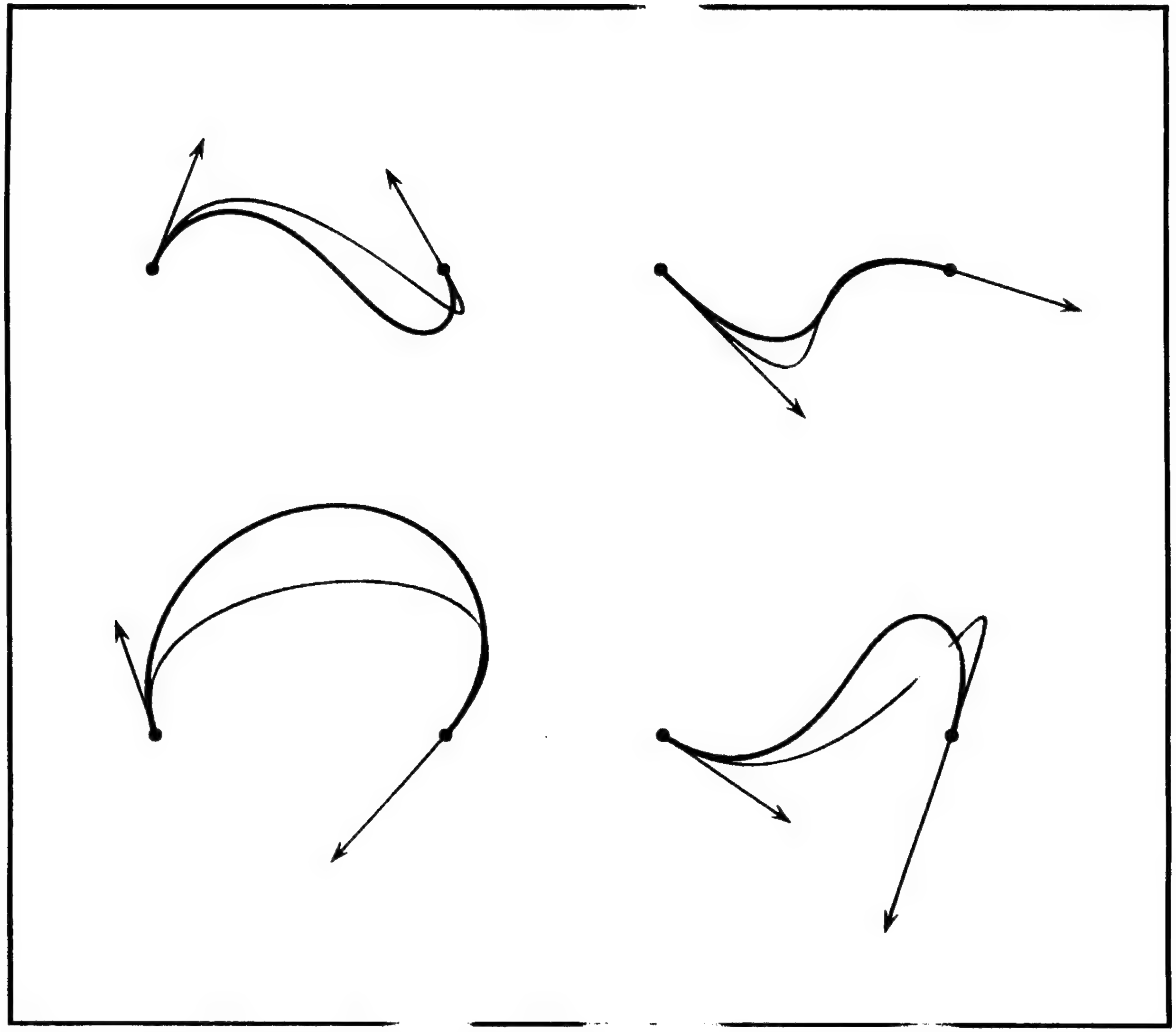


V
(a)
(b)
(c)
s/S
0
1
2
3
4
5
6
7
8
9
0.2
0.4
0.6
0.8
1

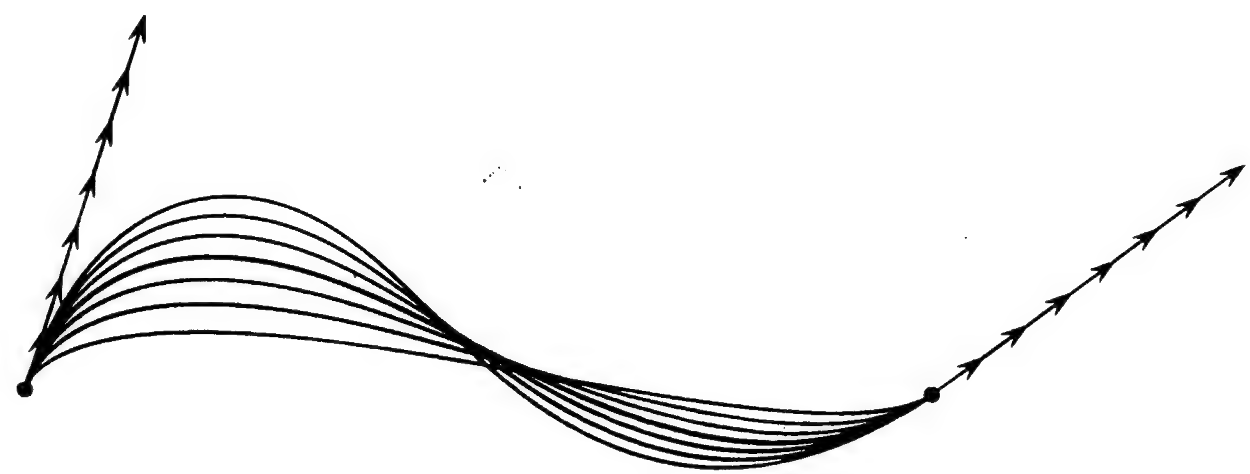
feedrate inversely proportional to curvature



feedrate	# cycles	time
constant	210	8.40 μs
linear arc length	236	9.44 μs
quadratic arc length	261	10.44 μs
inverse curvature	178	7.12 μs



PH quintics are more "graceful" than ordinary cubics



optimizing the bending energy of PH quintics

complex formulation

real and imaginary parts of the *square*
of a complex polynomial $\mathbf{w}(t) = u(t) + \mathrm{i}\, v(t)$
form a Pythagorean hodograph:

$$\begin{aligned} \mathbf{r}'(t) &= x'(t) + \mathrm{i}\, y'(t) = \mathbf{w}^2(t) \\ &= u^2(t) - v^2(t) + \mathrm{i}\, 2\, u(t) v(t) \end{aligned}$$

write hodographs of adjacent PH quintics as

$$\mathbf{r}'_k(t) = [\, \mathbf{a}_{k-1} (1-t)^2 + \mathbf{b}_k\, 2(1-t)t + \mathbf{a}_k\, t^2 \,]^2$$

$$\mathbf{r}'_{k+1}(t) = [\, \mathbf{a}_k (1-t)^2 + \mathbf{b}_{k+1}\, 2(1-t)t + \mathbf{a}_{k+1}\, t^2 \,]^2$$

$\rightarrow$ automatically satisfy C^1 condition

$$2\,\mathbf{a}_k = \mathbf{b}_k + \mathbf{b}_{k+1}$$

C^2 continuity at nodes $k = 1, \ldots, N-1$

$$\begin{aligned} 3\,\mathbf{a}_{k-1}^2 + \mathbf{a}_{k-1}\mathbf{a}_k + 3\,\mathbf{a}_k^2 + 3\,(\mathbf{a}_{k-1} + \mathbf{a}_k)\,\mathbf{b}_k + 2\,\mathbf{b}_k^2 \\ = 15\,(\mathbf{p}_k - \mathbf{p}_{k-1}) \end{aligned}$$

PH condition for arcs $k = 1, \ldots, N$

assume end–derivatives $\mathbf{d}_0$ and $\mathbf{d}_N$ given:

$$\mathbf{a}_0^2 = \mathbf{d}_0 \quad \text{and} \quad \mathbf{a}_N^2 = \mathbf{d}_N$$

linearity of C^2 conditions
allows elimination of $\mathbf{a}_1, \ldots, \mathbf{a}_{N-1}$

"tridiagonal" system of N quadratic equations in N complex unknowns

$$\begin{aligned}
\mathbf{f}_1(\mathbf{b}_1, \ldots, \mathbf{b}_N) &= 17\,\mathbf{b}_1^2 + 3\,\mathbf{b}_2^2 + 12\,\mathbf{b}_1\mathbf{b}_2 \\
&\quad + 14\,\mathbf{a}_0\mathbf{b}_1 + 2\mathbf{a}_0\mathbf{b}_2 + 12\,\mathbf{a}_0^2 \\
&\quad - 60\,(\mathbf{p}_1 - \mathbf{p}_0) = 0\,, \\
\mathbf{f}_j(\mathbf{b}_1, \ldots, \mathbf{b}_N) &= 3\,\mathbf{b}_{j-1}^2 + 27\,\mathbf{b}_j^2 + 3\,\mathbf{b}_{j+1}^2 \\
&\quad + 13\,\mathbf{b}_{j-1}\mathbf{b}_j + 13\,\mathbf{b}_j\mathbf{b}_{j+1} + \mathbf{b}_{j-1}\mathbf{b}_{j+1} \\
&\quad - 60\,(\mathbf{p}_j - \mathbf{p}_{j-1}) = 0 \qquad \text{for } j = 2, \ldots, N-1\,, \\
\mathbf{f}_N(\mathbf{b}_1, \ldots, \mathbf{b}_N) &= 17\,\mathbf{b}_N^2 + 3\,\mathbf{b}_{N-1}^2 + 12\,\mathbf{b}_N\mathbf{b}_{N-1} \\
&\quad + 14\,\mathbf{a}_N\mathbf{b}_N + 2\mathbf{a}_N\mathbf{b}_{N-1} + 12\,\mathbf{a}_N^2 \\
&\quad - 60\,(\mathbf{p}_N - \mathbf{p}_{N-1}) = 0
\end{aligned}$$

defines C^2 PH quintic spline

homotopy method

system to be solved:

$$\mathbf{f}_1(\mathbf{b}_1, \ldots, \mathbf{b}_N) = 0, \ \ldots, \ \mathbf{f}_N(\mathbf{b}_1, \ldots, \mathbf{b}_N) = 0$$

"initial" system with trivial solutions:

$$\mathbf{g}_1(\mathbf{b}_1, \ldots, \mathbf{b}_N) = 0, \ \ldots, \ \mathbf{g}_N(\mathbf{b}_1, \ldots, \mathbf{b}_N) = 0$$

"continuously deform" initial system into desired system
while tracking the motion of all solutions

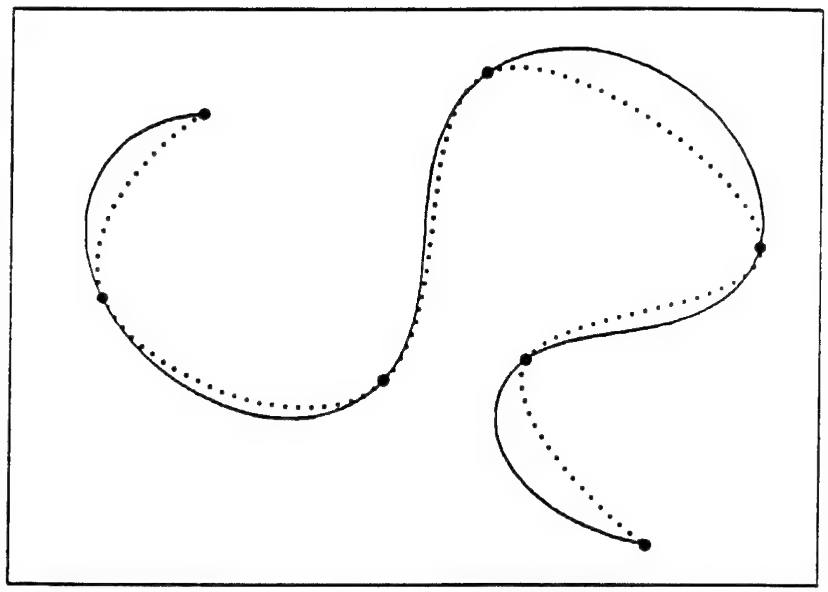
$$\mathbf{h}_j(\mathbf{b}_1, \ldots, \mathbf{b}_N, \lambda) = \lambda\, \mathbf{f}_j(\mathbf{b}_1, \ldots, \mathbf{b}_N) + (1 - \lambda)\, \mathbf{z}\, \mathbf{g}_j(\mathbf{b}_1, \ldots, \mathbf{b}_N) = 0$$

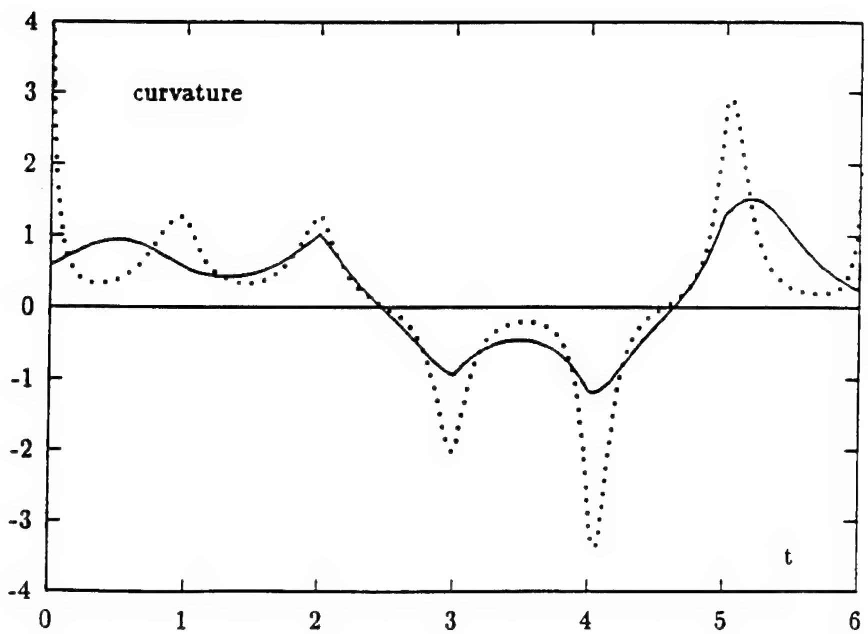
"homotopy parameter" $\lambda \in [0, 1]$

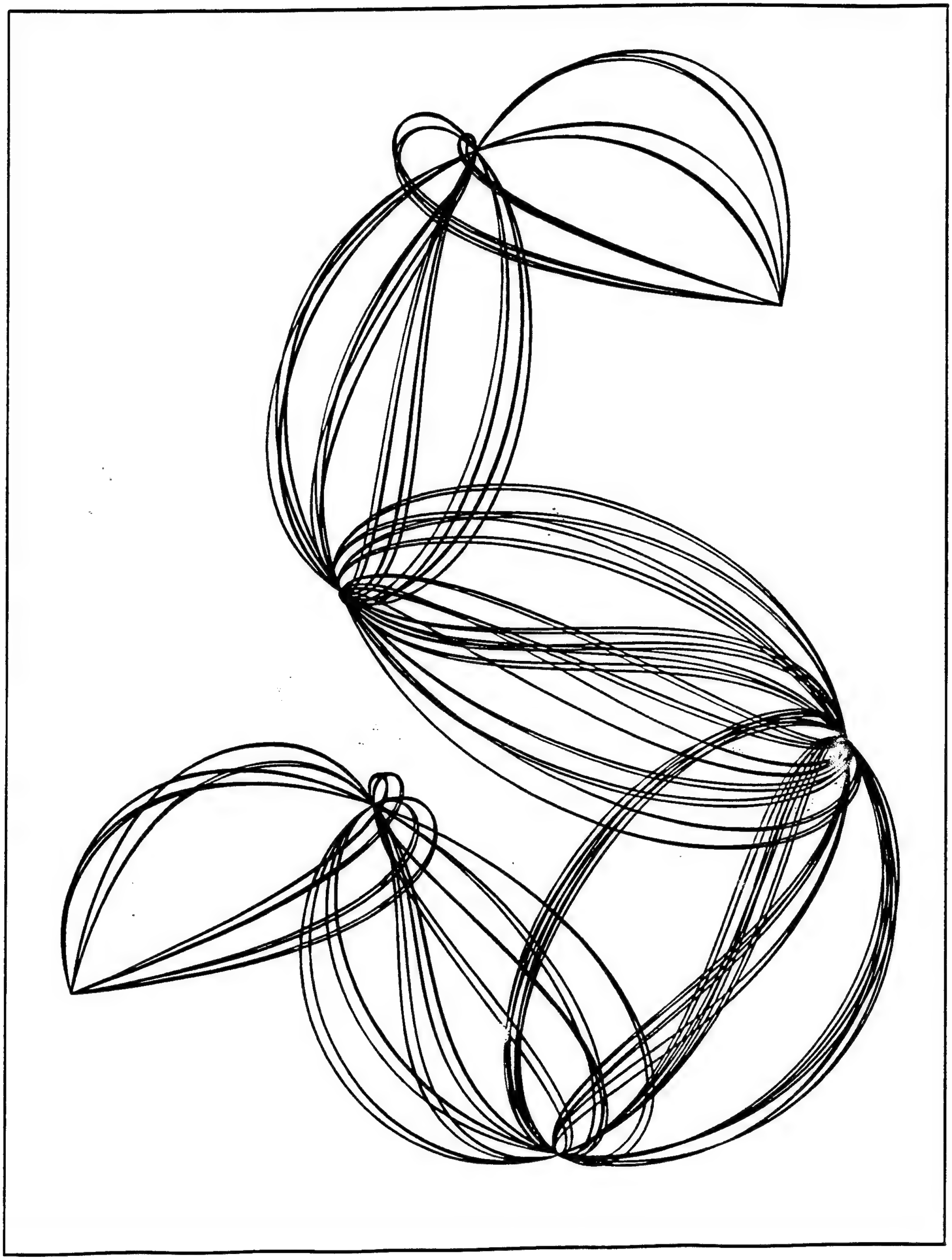
$$\mathbf{h}_1(\mathbf{b}_1, \ldots, \mathbf{b}_N, \lambda) = 0, \ \ldots, \ \mathbf{h}_N(\mathbf{b}_1, \ldots, \mathbf{b}_N, \lambda) = 0$$

homotopy system defines solution loci in $\mathbf{C}^N \times \mathbf{R}$
loci are non–singular for "almost all" $\mathbf{z}$

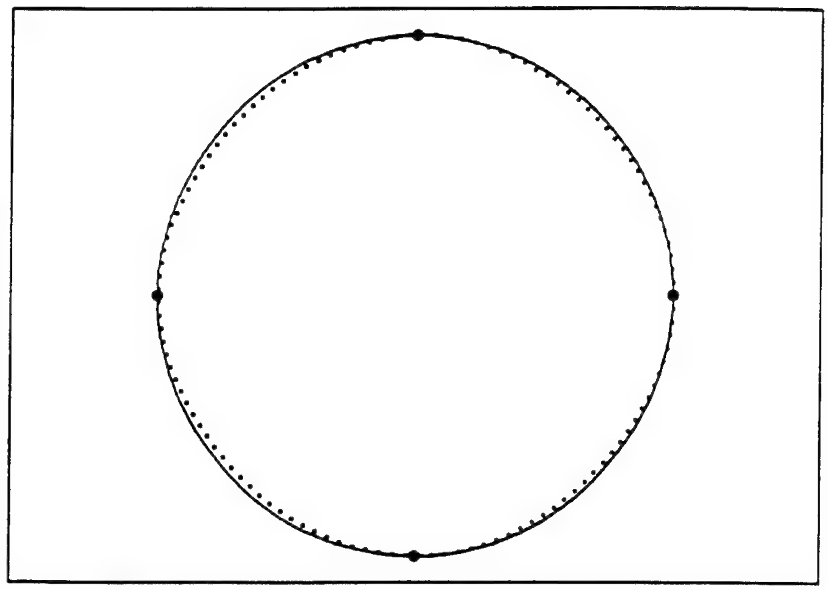
compare with "ordinary" cubic spline

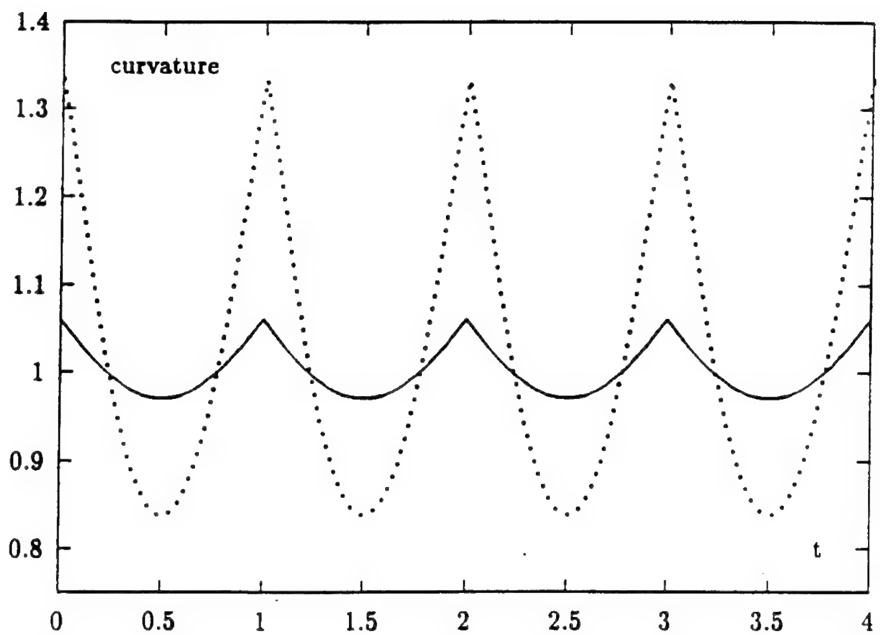






approximation of unit circle —
4 points & periodic end conditions





A Constraint Driven 'Piping Algorithm' Using Cyclides

Graduate Students: Seth Allen and Vinod Kumar
Faculty: Deba Dutta
Mechanical Engineering Dept.
Univ of Michigan, Ann Arbor, MI

Abstract

Cyclides are (tubular) algebraic surfaces of degree four. They are attractive for use in CAD environments for several reasons: dual representations, intuitive defining parameters, etc. Blending applications of cyclides have been demonstrated in the past. In this presentation, we will describe a procedure to compose cyclide pieces that satisfy designer imposed spatial constraints. This development is in the context of an important application: cable/wire harness design. We consider the wire harness problem and propose a two-step solution. First, generate a "configuration" that staisfies the logic diagram specified by the designer. Next, route the cable within the environment, for the given configuration. The constraint driven cyclide piecing technique is used in the second step.

Each wire bundle is represented as a cyclide tube (which in turn could be composed of several cyclide pieces) It is an interactive scheme in which the harness designer "guides" the cables (cyclide pieces) while several constraints are automatically satisfied. Examples of constraints include number of junction points in the configuration, max/min bend radii, max/min cable dia, max/min length of cable, regions in the environment to be avoided/included, etc. These constraints form an optimization problem and using Sequential Quadratic Programming, we obtain the collision free geometry of each cyclide piece.

The first step, i.e., configuration generation, is a graph theoretic problem and possible approaches include Steniner trees and algorithms for (minimum) spanning trees. Analogies to VLSI and network synthesis are being explored to determine a viable scheme for generating a good configuration. Note, for a given logic diagram, there exists a multitude of configurations and currently, in industry, no more than one configuration is considered. In terms of AF applications and relevance, we will discuss our problem in the context of harness design problems at Lockheed Martin (air craft design) and at Phillips Lab, Kirtland AFB (wire harness issues in space satellite design).

Work related to Phillips Lab, Kirtland AFB:

Description:

- Medium Wavelength InfraRed camera (MWIR) is attached to the Vibration Isolation and Suppression System (VISS) which in turns sits on the mounting plate on the satellite.
- The MWIR needs to be isolated and stabilized well in order to remove jitters (which would blur the images). The VISS would provide the stabilization to MWIR.
- There are wires (approx. 82) which run from MWIR to the electronic box on the payload. These wires bypass the VISS.

Considerations (contd.):

- Using more cyclide pieces and shorter step lengths make it easier to pick good initial values.

Future Work:

- Additional constraints:
  - Weight
  - Stiffness
  - Cost
- Automatic generation of good junction points
- Cost estimation of preposed route to aid in automatic generation of configurations

Problems:

- The wires adds considerable stiffness to the VISS which needs to be mechanically "soft".
- The wires possess non-linear characteristics such as "stiction" from insulation on wires rubbing against each other.
- The routing geometry of the wires affect their stiffness and load distribution (weight) on VISS and mounting plate.

Problem Statement:

Design and route a harness which minimizes the stiffness, non-linearities and weight of the MWIR wires and meets the required isolation/stabilization goals.

324

The Optimizer

The optimization code uses a standard optimization technique **Sequential Quadratic Programming:**

- Objective function is approximated by a quadratic function.
- Constraints are approximated by linear functions.
- The gradient of every constraint is found by finite difference methods.

We use the package **CFSQP** written by C. Lawrence, J. Zhou, and A. Tits from the University of Maryland.

CFSQP first takes the user specified initial values to generate a feasible point. Then each iteration finds a new feasible point that reduces the value of the objective function.

Discussion

Running Times:

- Optimization Per Piece:
  Good initial Values: ~30 seconds
  Bad initial Values: 1-5 minutes (may not find feasible point)
- Total Time for Lockheed Example: 2.5 hours

Considerations:

- Allowing bulges (extra free space around cable) improves time required for optimization.
- CFSQP often has trouble finding a feasible point, so good initial values are important.

Optimization Step

The initial information chosen by the user is sent to an optimization program that finds the best cyclide that

- satisfies the set of constraints, and
- connects the start position to a final position as near the selected end position as possible.

Seven variables in the optimization (per piece)

- the vertex of the tangent cone (3),
- the center of the specified final position (3), and
- the radius of the final sphere (1).

The Optimization Problem:

Minimize $\mathcal{F}$ subject to:

- $a \leq \min(r) \leq \max(r) \leq b$,
- $c \leq \min(R) \leq \max(R) \leq d$,
- $e \leq L \leq f$,
- $D_i > 0$, $i = 1, 2, 3, \ldots,$ number of obstacles

Where:

$\mathcal{F}$ is the distance from the final sphere in the cyclide to the user selected goal.

r is the range for the internal radius of the cyclide.

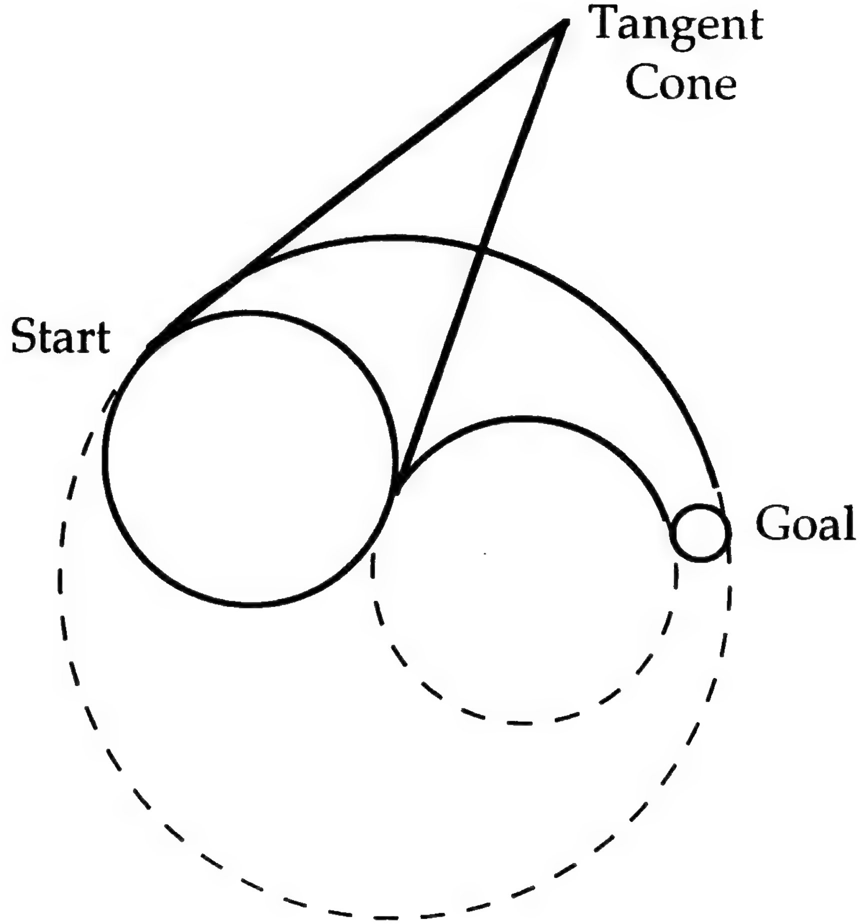
R is the range for the bend radius of the cyclide.

L is the length of the cyclide along the spine curve.

D_i is the distance from the cyclide to the ith obstacle.

a, b, c, d, e, and f are user specified.

Interactive Step (Contd.)

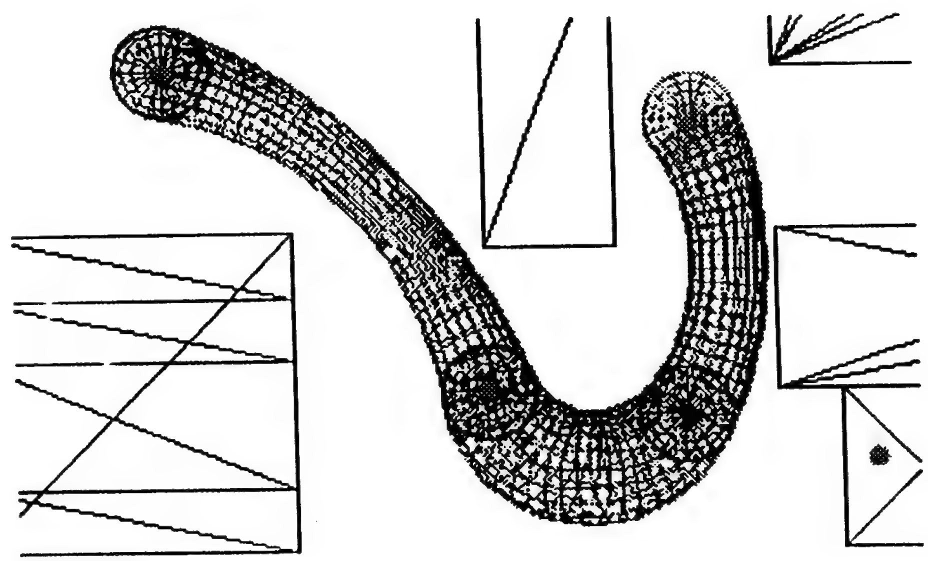


Constraints Imposed by the Designer:

- Internal radius of the cable must lie between a specified maximum and minimum internal radius.
- Bend radius of the cable must lie between a specified maximum and minimum bend radius.
- The length of the cable must lie between a specified minimum and maximum length.
- The cable must not intersect any obstacles.

Method

Interactively construct tube piece by piece where the designer modifies/selects initial values which are then optimized to ensure a valid solution.



Interactive Step

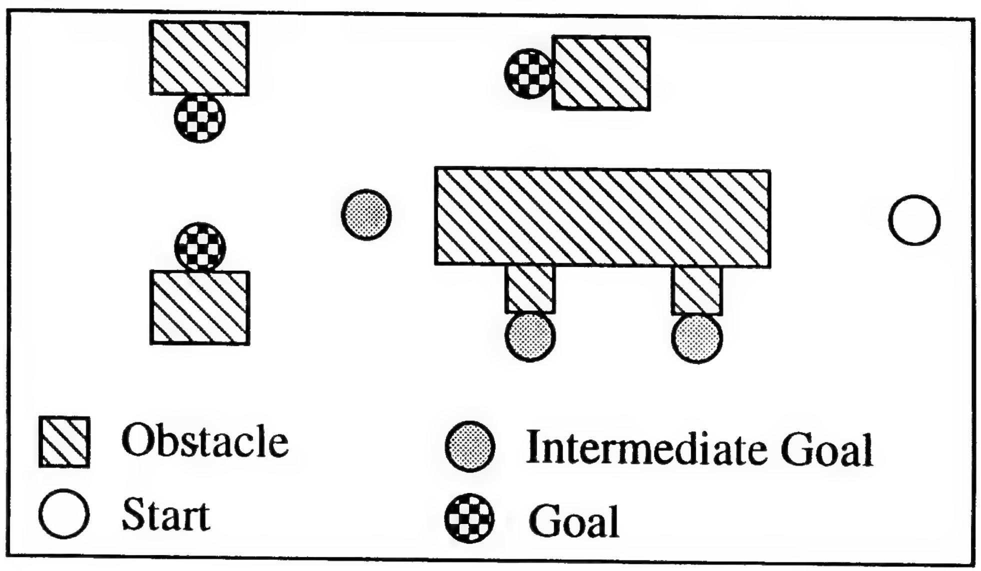
Construct a single piece of the cable.

User Selects:

- start position (either end of previous piece or from list of start positions)
- end position (either from list of goal positions, or a point that moves cable nearer some goal)
- an initial tangent cone (direction) for this piece of the cable

Stage 2: Collision Free Cable Routing

Problem: Connect a given start position to a given goal position with a wire that avoids obstacles and satisfies a given set of constraints.



Approach to Routing Problem

Use cyclides to represent the free space in which the cable will be placed.

Input:

- Set of start positions,
- Set of goal positions,
- Obstacles, and
- Intermediate goals (which indicate where cable splits, joins another cable, passes through a clamp, etc.)

Configuration design contd.

- Each configuration could be one of the following:
  - A spanning tree if there are no junctions introduced. It could be optimized based on some cost function to generate a minimum spanning tree.
  - If there are junction points in the configuration, it is a Steiner Tree. More specifically, an Euclidean Steiner Tree. Again, it could be optimized to obtain a Euclidean Steiner Minimal Tree. It is known that this problem is NP-hard.

Ongoing Work:

- We have identified that the configuration problem as a variant of "The Steiner Tree problem" — An Euclidean Steiner Minimal Tree avoiding Obstacles (ESMTO).
- Approaches (mostly heuristics) exist to generate Steiner trees. We are looking into some of them which can be modified for the cable harness design.

Graph theory definitions (contd.)

- The spanning tree with the smallest weight in a weight graph is called a ***minimal spanning tree.*** (MST)

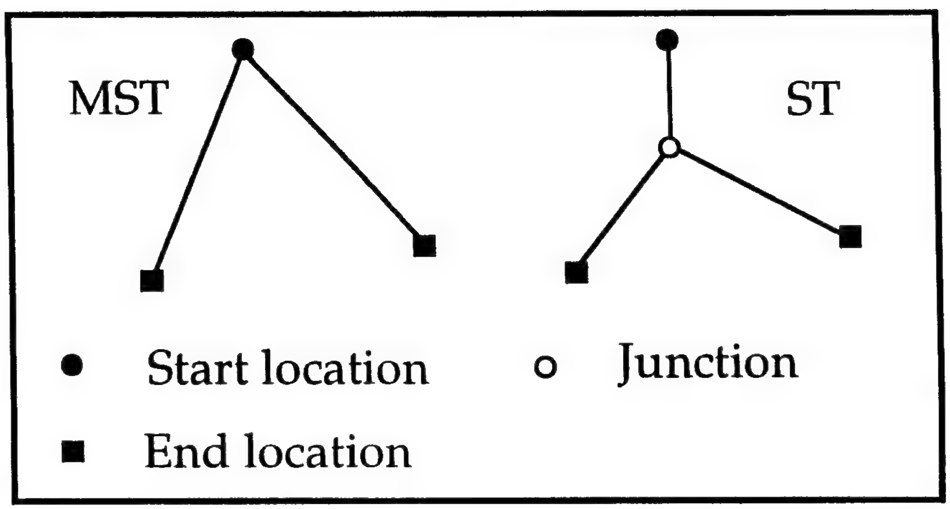
- A ***Steiner Tree*** (ST) for a given vertex set Z (terminals i.e. start/end points) is defined as the tree $T = (V, E, c)$, $Z \subseteq V$. Each vertex in Z has a degree 1 and $V-Z$ is the set of intermediate nodes called *Steiner points* (junctions).

- A Steiner tree is called an ***Euclidean Steiner Tree*** (EST) if the weight of each edge in ST is the Euclidean distance between its nodes.

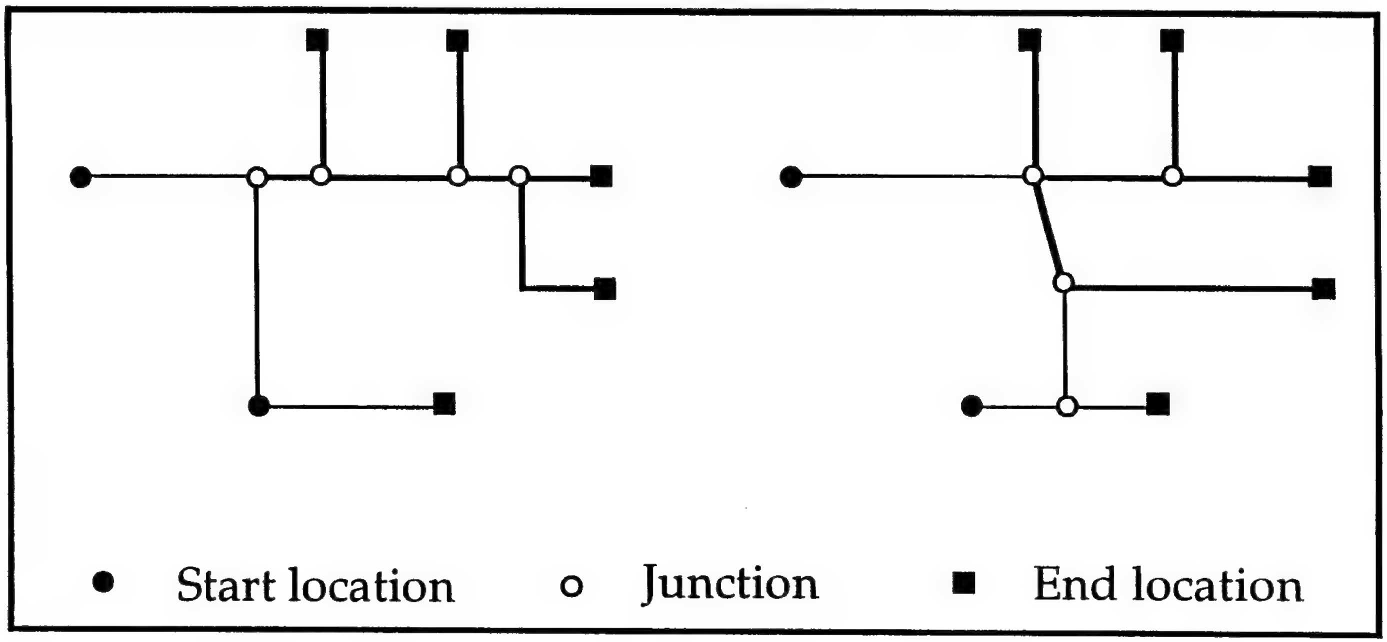
- If the Steiner tree has a minimum weight, its called a ***Steiner Minimal Tree*** (SMT). For EST, the minimal tree is the ***Euclidean Steiner Minimal Tree*** (ESMT).

Variants of the Steiner Tree problem — Applications:

- Wire routing phase in physical VLSI design.
- Networks: Water or electricity supply, communications, offshore oil pumping, etc.
- Design of transportation systems



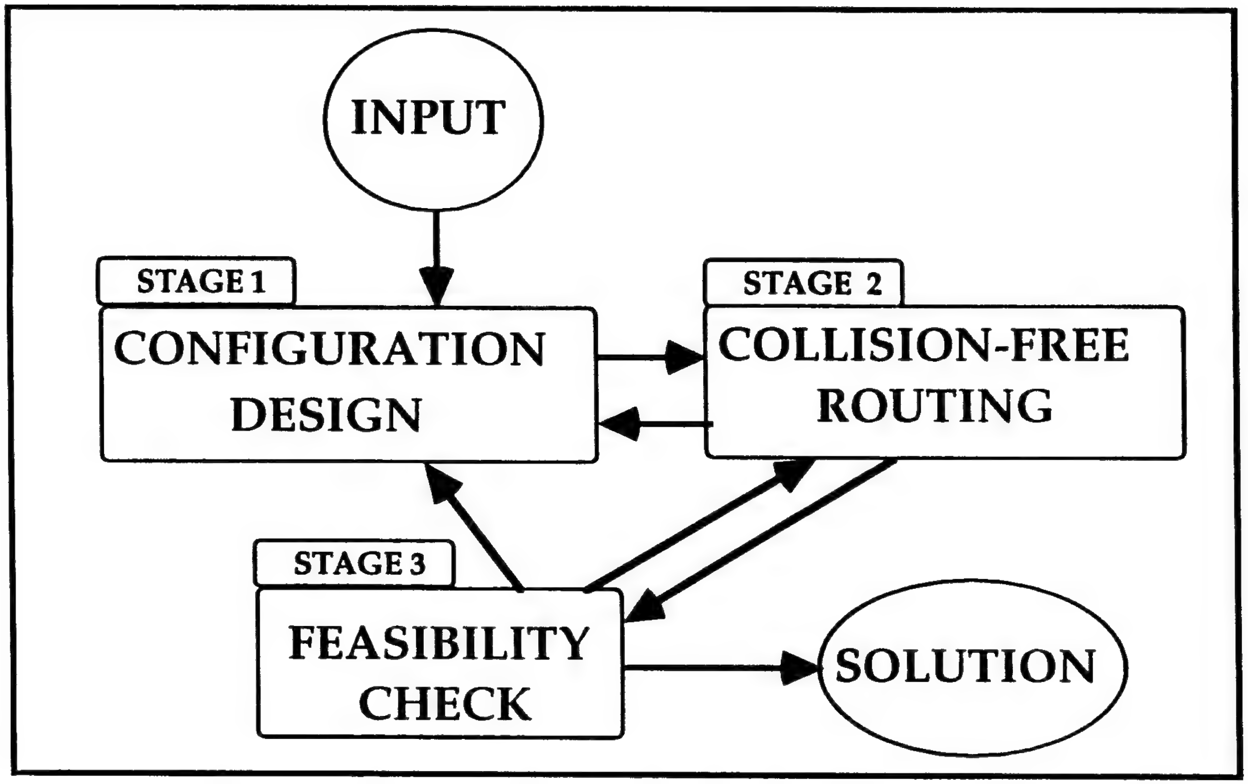
Two configurations for the same set of start & end locations



Graph theory definitions:

- ***Graph*** G = (*V*, *E*) consists of a vertex set V and an edge set E. Each edge is identified by an unordered pair of two vertices.
- A ***weighted graph***, G = (*V*, *E*, c) is a graph where c: $E \longrightarrow R$ is an edge function which associates each edge with a real number called the *weight*. All edges of an ordinary graph have weight one.
- A ***tree*** T is a connected graph with no loops.
- A tree T is a ***spanning tree*** of a connected graph G if T is a subgraph of G and T contains all vertices of G. Also called the skeleton.
- If G is weighted, then weight of a spanning tree of G is sum of the weights of all branches in T.

Overview of the solution strategy:



Stage 1: Configuration Design

- Configuration defines the topological structure of the harness. It identifies the way in which various locations are to be connected.
- ...ct geometrical positions of the start and end points are not considered.
- For a given set of starting and ending points, several configurations can be constructed.
- A configuration can also contain intermediate points (junction) where the harness can split or merge.

Objective (contd.):

- The system should address various issues like:
  - Configuration/ topological structure of the harness.
  - Collision free routing of each segment (bundle) of the harness.
  - Geometrical constraints like minimum/ maximum bend radius, internal radius, length etc.
  - Material properties like weight, stiffness of the cable should be considered.
  - Case-specific constraints like locations of clamps etc.

Proposed Solution Strategy:

A three stage methodology is followed:

STAGE 1: Generation of the configuration.

STAGE 2: For each configuration generated in stage 1:

- Generation of collision free paths for each segment in the configuration which also satisfy all the imposed constraints. An optimization scheme is used to get a local optimal solution for each segment (bundle).

STAGE 3: Do a feasibility check on the generated solution — if it satisfies all the designer's requirements.

Our Philosophy:

Solution:

- Impractical to obtain a global minimum for the problem.
- Computational costs will be prohibitive. Hence, full automation of the harness design not practical.
- A better approach is to keep the user/designer involved, making it an interactive system.

34/5

Objective:

Develop a CAD-based wire harness design tool which enables:

- the harness design to be addressed in early stages of the design.
- Automate as many steps as possible, and the make the rest interactive.
- Generate near optimal (i.e. good), solutions rapidly as several iterations might be required before a final solution is selected.

346

Cable Harness Design

Problem:

Design and route cable harnesses in complex 3-D environments like interior panels of automobiles, aircrafts etc.

Background:

- Harness is used to house wires which connect various components in the assembly.
- Very labor-intensive and time consuming process.
- Usually done in the last stages of the product design.
- Often subjected to change due to change in component geometry and their placement in assembly.

Unconstrained cyclide tube composition (contd.)

- The cyclide tube is represented by two end spheres and a tangent direction at one end. The tangent at the other end is automatically determined.
- If there is no tangent continuity between two adjacent tubes, a part of the common sphere might be needed to complete the surface.

Applications of unconstrained cyclide tube composition:

- Synthesis of geometrically complex shapes
- Pipe joining
- Free volume modeling

Constrained cyclide tube composition scheme

- Impose constraints on all parameters of the cyclide (locations and radii of the spheres, tangent cone direction, extreme circles' radii etc.) based on the design application.
- Generate the cyclide tubes (whose parameters are constrained) such that it maximizes/minimizes certain objective function.

Application:

Cable Harness design : The free space available for cables is modeled as cyclide tubes so that they satisfy all the design constraints/requirements and avoid collisions with parts.

Solution Strategy:

STEP 1: As the part is axisymmetric, position the probe (using z and ϕ) so that its mid-plane lies in the plane containing the axis of the part and the goal point.

STEP 2: Coarsely divide the C-space (space containing all possible configurations - x, y, θ, h and α) into cells.

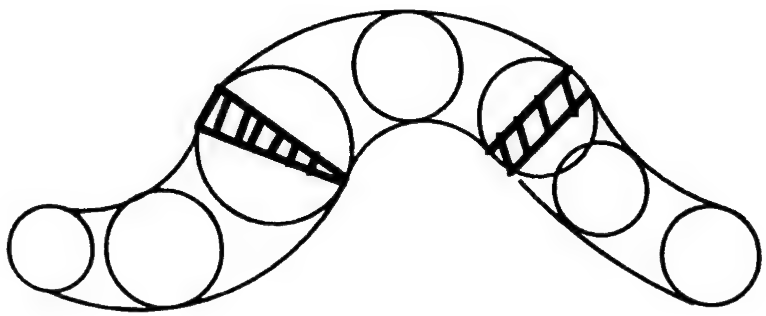
STEP 3: Identify the free space, the initial C-cell (home position) and all C-cells for which the probe reaches near the goal point.

STEP 4: For each final C-cell, use wavefront algorithm to find a path (if it exists) from final C-cell to the initial C-cell such that α is held constant. Note that all α values are permissible at initial position.

STEP 5: Divide each final C-cell (having a path from initial C-cell)- finely and repeat steps 3 & 4 to exactly reach the goal point.

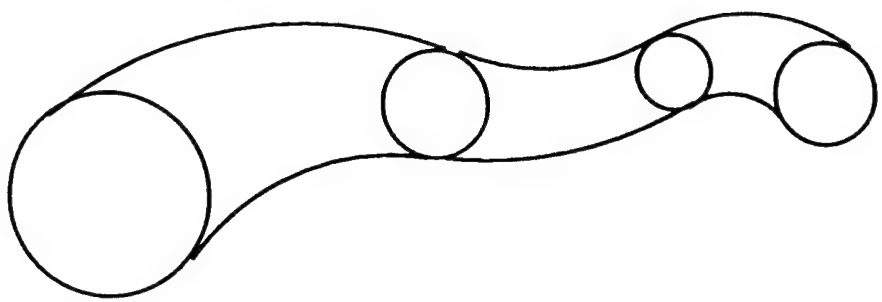
Unconstrained Cyclide Tube composition

(previous work)



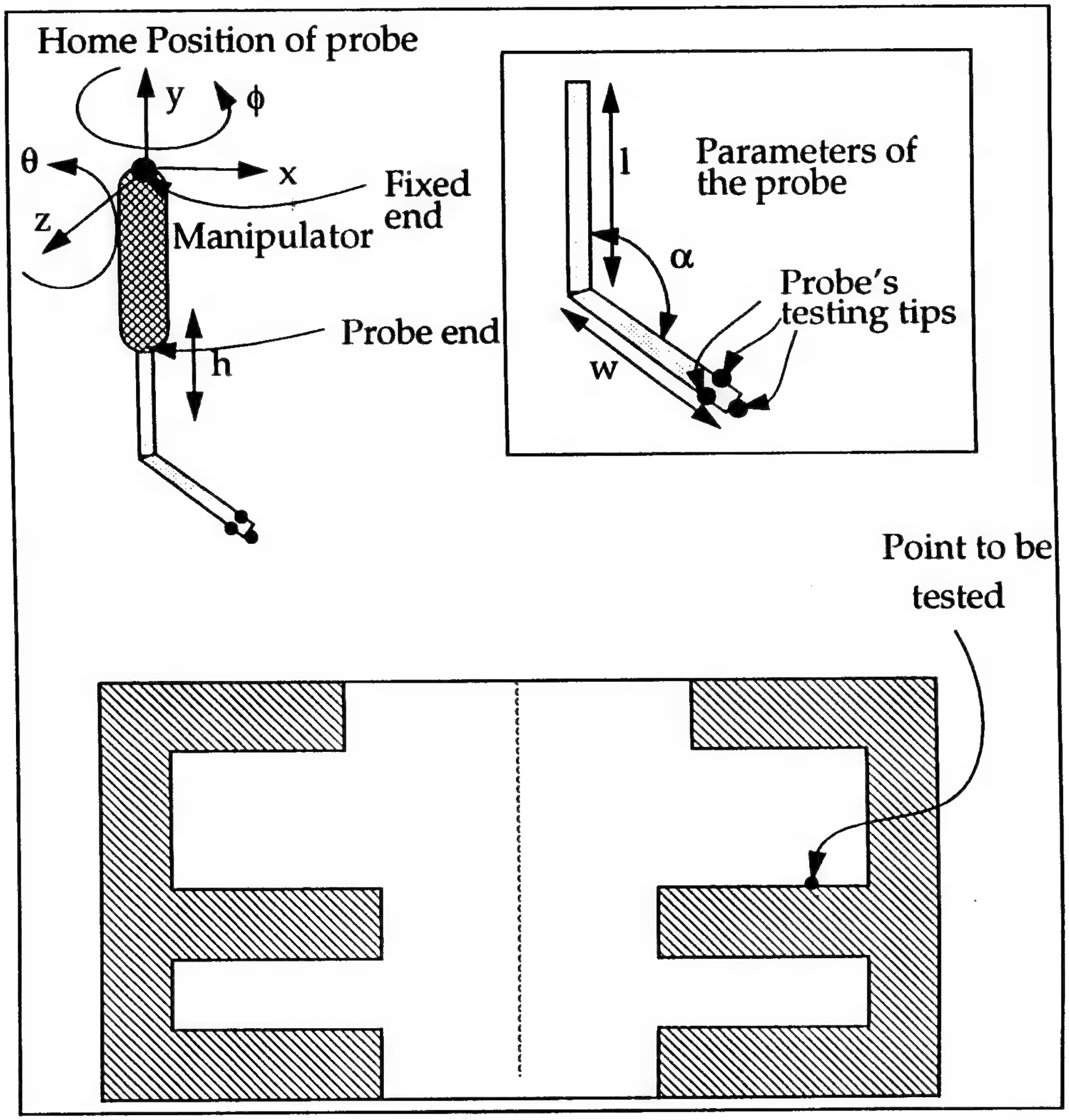
Unconstrained cyclide tube composition

Impose Tangent continuity constraint between two adjacent tubes:



Tangent continuous cyclide tube composition

EDDY CURRENT PROBE SETUP:



Problem Statement:

- Move the eddy current probe from the home position such that one of the testing tip touches the final point on the part.
- Identify a suitable geometry of the probe.

Input to the problem

- Home position of the probe, position of the part and the goal point in global reference frame.
- Geometry of the manipulator arm:
  - The value of w is set to 0.3-0.5 times the inner diameter of part.
  - The value of l is not critical as there is a degree of freedom along its length. Its value is set to 1.2w.
  - The angle α is set as a degree of freedom and its value identified by the algorithm.

354

Degrees of Freedom:

- Choice of cone or cylinder
- Height of cone/cylinder
- Diameter of cone/cylinder
- Number of sides
- Angle between sides
- Circle on cone/cylinder where cyclides meet quadric

Properties of Corner Blends

- Uses 2n cyclide patches, including n four-sided patches and n triangular patches.
- Cone/Cylinder must be in symmetric position relative to each side of the corner.
- Corner must be symmetric; i.e., angle at vertex on each side must be equal.

The Cyclide

Parameters: a, c, m (b is given by $\sqrt{a^2-c^2}$)

Implicit Forms:

$$(x^2+y^2+z^2-m^2+b^2)^2 = 4(ax-cm)^2 + 4b^2y^2$$

$$(x^2+y^2+z^2-m^2-b^2)^2 = 4(cx-am)^2 + 4b^2z^2$$

Parametric Form: $(0 \leq \theta, \Psi \leq 2\pi)$

$$x = \frac{m(c - a\cos\theta\cos\Psi) + b^2\cos\theta}{a - c\cos\theta\cos\Psi}$$

$$y = \frac{b\sin\theta\,(a - m\cos\Psi)}{a - c\cos\theta\cos\Psi}$$

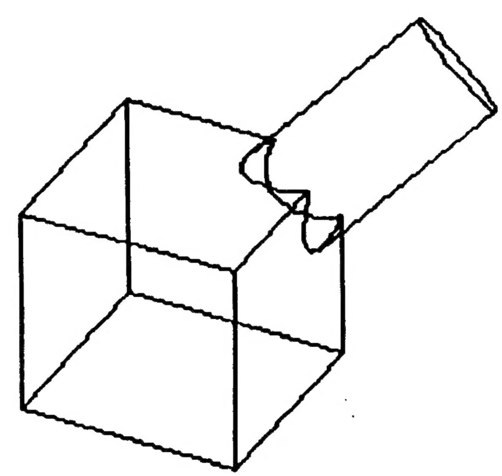
$$z = \frac{b\sin\Psi\,(c\cos\theta - m)}{a - c\cos\theta\cos\Psi}$$

356

Symmetric Corner Blends

Blend between

- a cone or cylinder (or sphere), and
- an **n**-sided symmetric corner where n = 2, 3, 4, ...



358

AFOSR Review Meeting

June 1994 - May 1995

Graduate students:

Seth Allen
Vinod Kumar

Faculty:

Deba Dutta

Department of Mechanical Engg. & Applied Mechanics
Design Laboratory
University of Michigan
Ann Arbor, MI 48108

May 24, 1995

359

REVIEW TOPICS

- Cyclide Blends
- Path planning for eddy current test probes
- Constraint driven cyclide tube composition scheme for wire harness design

Forum Discussion

	Priority	Applications
Discrete Event Design Analysis		
Simultaneous Design & Analysis		
In-situ Process Design & Discovery		
Computationally Efficient Algorithms		

3<o0